产品技术手册

高可用产品技术白皮书

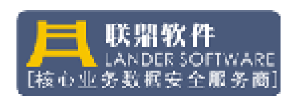
LanderVault 集群模块

LanderCluster 6

本手册简明阐述了系统及方案基本特性与说明

www.landersoft.com

www.landercluster.com



目录

1 集群(高可用)概述

在学习使用 LanderCluster 集群软件包之前，必须对集群技术中的一些概念有所了解，这样才能帮助我们更好的掌握 LanderCluster 的原理、安装设置和管理。

这一部分，我们将掌握下面列出的部分内容：

A 什么是高可用(High Availability)

B 高可用是哪些业务环境的需求

C 高可用的标准

D 实现高可用的难点

E 高可用系统实现的准备

F 高可用系统准备的关键点

G 高可用实现

1.1 高可用的基本概念

1.1.1高可用技术中的几个术语

在我们学习高可用概念前，先定义一些术语， 就像 'Availability', 'High Availability', 'High Available Computing'等。

A、可用性 (Availability)：

是指按照需要提供一定级别服务的系统。这个概念是体现在我们生活、工作中，在计算机领域，可用性通常看成是系统提供服务的时间段(如一天 16 小时, 一周 5 天)或是系统的响应时间(如:1 秒钟的响应时间)。任何的服务丢失，包括计划中或计划外（意外）的，都被定义为损耗(OUTAGE)。宕机时间(Downtime)是指系统从停止服务到重新提供服务的时间(以时间单位计量，如分、小时、天等)。

B、高可用性（High Available）：

High Available 指定义一个系统，使之能够通过减少或对错误的控制以降低系统宕机时间来尽可能避免服务丢失。我们能够健康、快乐的生活、公司的正常运作，这些需要能够有一个安全的、可靠的环境作为保障。

例如，我们希望供电系统可靠，哪怕一点点、短暂的停电都是不可接受的，因为我们的生活已经离不开电力了，像冰箱、空调、微波炉、照明等，停电意味着生活无法正常进行。

甚至当非常可靠的服务突然非正常停止，我们还是非常希望能够马上恢复。 当供电系统不正常时，我们期望电力公司的抢修车能够以最快的速度修好。

C、高可用计算

（High Available Computing）：

在一些业务系统中,计算机的可靠性几乎同供电系统的可靠性具有一样重要性。 高可用计算(High Available Computing) 是被设计成只容许有极短的计划和非计划宕机时间的计算机系统。

需要说明的是高可用(High Availability)也不是绝对的，不同的业务系统对高可用的需求是不一样的。 例如银行的信用卡系统、轮班制的企业(24 小时不停的流水线)和一些提供服务的网站，要求系统 24 小时不停止运行； 一些金融单位(证券交易所)系统要求一周 5 天，每天白天或夜间交易时间内不能停机，其它时间可以停机做维护等；同时一些零售企业（商场）仅仅需要系统每天运行 18 小时，但是要求具有很短的响应时间来进行事物处理。

D、服务级别（Service Levels）：

系统的 Service Level 是指系统提供给用户的服务级别。通常，服务级别在有关专业技术文档中有相关描述，但并非十分严格。这里可以简单的理解为：服务级别是对提供服务的系统服务能力的量化。

高可用环境可以提供一个服务级别的服务，使得系统的计划及计划外宕机时间不超过一个特定的时间。

E、连续可用（Continuous Availability）：

“连续可用”意味着永不停止的服务，包括计划内和意外服务的终止。这是一个比高可用要求更加难于实现的环境，意味着服务不能够有任何意外发生。实际上连续可用的系统在现实中是不可能存在的。

因此这个概念通常指运行的系统要求只能有极少的服务终止时间，即指非常高的可用系统。高可用不意味着就是连续可用。

F、容错 (Fault Recovery)：

容错系统不是可用性级别中定义的一种，而是实现更高级别可用性的方法。容错系统是硬件冗余的概念，该系

统通常使大多数部件硬件冗余，包括 CPU、MEM、I/O 系统及其他部件。容错系统能够保证在硬件、软件出现故障时，系统可以继续提供服务。但是容错系统不能避免人为失误造成的服务终止。高可用系统同样不意味着容错。这样的系统像 Stratus 这样的厂商提供的产品就是容错机产品。

G、容灾 (Disaster Recovery):

容灾系统是可以容许灾难发生的运行系统。可以容许系统发生多点故障甚至整个系统损坏的情况下，服务不受影响。通常，容灾系统中的服务器分别运行在不同地点，通过网络线路保持数据的一致，在运行主机故障或灾难发生时，其他容灾服务器可以接管其服务。容灾系统中服务器节点可能分布在校园不同的建筑内、城市的不同区域、甚至跨越海洋，在不同的洲运行，可见这样的系统需要大量的投资和大量的维护工作。只有非常关键的业务系统会采用这种方式保证数据的安全和系统的可用。

H、5nines:5minutes(五个九：五分钟):

早在 1998 年，惠普针对其推出的服务器系统提出了 99.999%可用性，指的是系统的宕机时间一年不会超过 5 分钟。这个定义使系统的可用性、可靠性具有一定的数据参考的标准，同时大大促进了集群技术的发展。

不是所有的用户要求所有的设备和工具提供 99.999%的可用。不是所有的用户愿意有这样的投入。但是所有的用户都在可用性技术的发展中获益。比如你不希望家里的轿车具有赛车的引擎，但是赛车的引擎技术发展的确促进了家庭轿车引擎的进步。

1.1.2分析用户的可用性需求

非正常的服务终止时间的长短会对用户造成不同的损失，或者说用户对服务停止所能承受的时间是不同的。通常取决于应用的类型，如在一秒钟内修复错误，不会对一个在线联机事务(OLTP)处理系统构成影响，但是对于一个科学计算应用运行在实时环境下，则停止哪怕一秒都是不可忍受的。

由于系统的任何一个部件都可能发生故障，因此挑战是在设计系统时能够预判哪些故障将要发生，并且能够最快的纠正将要发生的错误。

1.1.3选择一个解决方案

用户系统可用性的要求决定方案的选择。例如：如果系统停机多个小时不会影响业务，那样你就不需要购买带有热插拔硬盘的存储系统。另外如果你不能忍受硬盘更换造成的停机，则可以选择用热插拔的磁盘阵列、并且可以通过硬盘的 Mirror(镜像)达到硬件容错效果。

我们关心的是基于各种操作系统环境的高可用系统，因为在 PC 服务器硬件环境被广泛采用的情况下，更多具有高可用要求的系统在运行，那么如何更好的满足用户的需求至关重要。LanderCluster 的高可用技术完全来自于联鼎（Lander SoftWare）技术人员对高可用系统多年的研究，对用户系统，特别是政府、银行、证券、邮政、保险等行业用户的应用分析而研制开发的。、

1.2 高可用是业务的需求

目前的很多业务，高可用系统是实实在在的需求，而不是华而不实的概念了。从某种意义上讲，高可用系统是对系统宕机造成数据丢失的一种保障；从另外一点看，通过它，企业可以为用户提供更好、更具竞争力的服务，增强了企业的竞争力。

1.2.1高可用是一种保障

高可用系统在以下损害情况下，提供了系统的保障：

A 收入减少

B 客户不满意

C 丧失机会

对于商业计算，高可用方案是必需的，因为丢失系统服务意味着利润的损失。对于这样的业务，我们通常称之为关键业务（mission-critical）,对于所有的关键业务，系统宕机意味着收入的减少，高可用是必要的。对于银行，如自动取款机 24 小时提供服务，其应用系统是典型的关键业务。对于一些像证券交易这样有着安全需求的关键业务，高可用环境保证系统在交易时间内不停机运行，在交易结束后，可以将服务器关闭。

1.2.2高可用存在商机

由于不断增长的关键业务需求，为高可用计算提供了一个，包括金融、通信、能源、政府等各种行业都有不同程度的高可用需求。很难有严格的定义来确定一个系统是否为高可用系统，这些都基于实际的业务。一些业务是基于计算机的，下面的说法通常是正确的：

A　可以有很多方式实现高可用；

B　业务的可用性需求决定系统可用性配置，没有一个绝对的标准对所有的业务都适用；

C　高可用的实现方法影响整个系统的设置；

D　通过创建合理的操作流程和保护措施来减少可能的停机；

E　恢复必须是按照计划进行的；

下面的内容是我们期望运行在关键业务环境下的系统所能够达到的效果：

A　出错的概率应该很低，同时出错的间隔应该尽量的长；

B　应用应该具有出错恢复功能；

C　系统可以在线设置（不需宕机）；

D　应该有很短的计划停机时间；

E　系统管理工具必须可用，完整。

1.3　高可用系统的衡量标准

可用性和可靠性都可以用数字来衡量，当然这样也可能误导用户。

事实上，还没有一个标准来衡量一个计算机系统的可用程度。关键是我们可以定义一组数字，分别定义不同级别的可用系统。我们要记住的是，如同计算机中的 CPU 主频一样，可用性也不是一个可以直接衡量的属性。可用性只能用历史的眼光衡量或评估，只有一个系统实际运行多少时间后，才可以回头来衡量其可用的程度。另外，衡量系统可用，不能简单的问：“服务正常吗？”，而是“是否整个系统都在提供特定级别的服务？”。

同时，可用性同可靠性是相互联系的，但它们又是不同的概念。可用性指系统实际提供服务的时间和整个系统应该运行时间的比率（例如，一个系统可以达到 99.999%）,而可靠性是指系统从运行到出错的时间段。应该说可用性包含可靠性。

1.3.1可用性计算

可用性计算用以下公式得到：

可用性=实际运行时间/应运行时间（运行时间+宕机时间）。

通常，可用性用一个比率表示，一般是小时同系统正常运行的周、月、年的比率。

1.3.2期望运行时间

衡量系统的可用性必需结合企业的业务背景及对系统提供服务的要求。下面两个表格可以了解系统实际运行时间、宕机时间及不同可用性值。

表一是 7X24X365 的环境，表示系统运行在一天 24 小时、每周 7 天、一年 365 天。则其运行和宕机情况表：

单位（小时）

可用性	最少运行时间	最大可以宕机时间	剩余时间
99%	8672	88	0
99.9%	8716	44	0
99.95%	8755	5	0
100%	8760	0	0

通过上表的分析，该系统没有剩余的时间，一年 8760 个小时全部被占用，也就是说所有的系统维护工作必须在系统运行或在可容许的宕机时间内进行。因此，越高的可用性比率，代表容许系统可以出错的时间越短。

表二是 12X5X52 的系统，表示每天运行 12 小时、每周 5 天，每年 52 周的系统。例如这样的系统就是 12X5X52: 每天早上 8 点到晚上 8 点，周一到周五。

单位（小时）

可用性	最少运行时间	最大可以宕机时间	剩余时间
99%	3088	32	5642
99.9%	3014	16	5642
99.95%	3118	2	5642
100%	3120	0	5642

上述表格可以看出，一个 12X5X52 的系统，一年有 5642

小时的空余时间（剩余时间），在这些时间内，可以进行系统维护、规划等操作。即使在这样的环境下，非计划宕机也必须认真控制。

1.3.3计算平均故障间隔时间

可用性同系统的部件出错率有关。一个常用的衡量设备的可靠性尺度是平均故障间隔时间（MTBF）,这个尺度通常针对系统中的不同部件，如磁盘等。这些衡量尺度是非常有用的，但他们仅仅能代表高可用系统中的一个指标，例如，他们无法给出出错后，系统恢复时间的差异。

MTBF= Mean Time Between Failure

MTBF 的计算是累加所有部件（包括未出错部件）的实际运行时间除以系统中所有部件出错次数之和。实际运行时间指以小时计的系统运行时间之和（不包括关机时间）。

MTBF 用于表示部件或设备出错的平均时间间隔。通常的应用中，MTBF 用于通过对过去部件的性能情况来得到部件的期望性能。当该参数作为预测设备的可靠性时，它应该是一个稳定的值。

当多个同样部件同时工作时，计算其 MTBF 值是单个部件的 MTBF 除以部件的数量。意味着在多个相同部件同时工作，该设备的平均故障间隔会变小（出错概率增加）。例如一块硬盘的 MTBF 为 500000 小时，而 200 块硬盘在一个系统中的 MTBF 则只有 1000 小时，意味着每年可能的故障次数为 9 次。因此，我们可以得到一下结论：越多的部件同时工作会导致越多的系统出错概率。

1.4 实现高可用的难点

1.4.1高可用计算实现的难点

一个计算机服务的停止，通常被称之为一个 Outage，Outage 的时间称之为宕机时间（Downtime）,宕机时间分计划中和计划外两种情况。通常完成系统升级、应用迁移、部件更换等操作，计划中的宕机是必要的。

计划外宕机通常是由于系统出错造成的。错误包括硬件、软件、系统和网络，或是系统运行外部环境原因等，而人为失误造成的故障也称为出错（Failure）,并非所有的出错会造成 Outage,而且不是所有的意外宕机都是由于部件出错造成的，灾难或其他意外情况同样会造成服务终止。

1.4.2Duration of Outage (损耗时间)

衡量 Outage 情况的一个重要的信息是持续时间，针对不同的应用，一个 Outage 可能造成的影响不同。有些应用停止 10 秒没有问题，但是 2 小时将是非常严重的；有些应用甚至连 10 秒的停止都是不可容忍的。因此一个可用性系统必须能够满足停机时间小于应用的最低要求。比如一个 7X24 的关键业务要求做到 99.95%,则其宕机时间必须小于 5 小时/年。有些用户希望经常在每周、每月或每季度有计划的停机进行维护，这样有计划的停机将会降低系统非正常宕机的概率。

1.4.3系统损耗的时间分析

高可用的重要性可以从下面的一例子中得到答案，描述的是系统由于硬盘崩溃造成系统宕机的时间情况。下面是一个未使用镜像（Mirror）硬盘的 OLTP（联机事物处理）系统在硬盘损坏情况下的描述，当硬盘损坏时，整个系统将停止工作，直到硬盘修复。

未使用硬盘镜像时，正常情况下客户端可以连接并得到服务，在硬盘损坏时，整个服务器系统停止工作，客户端连接失败，在更换硬盘后，重新安装设置系统，启动服务，重新开始工作。这样的操作可能需要几个小时、甚至一天以上。这种情况下，恢复时间是无法预知的，同时这种情况是非正常宕机，不受控制。

当该系统的硬盘采用高可用方式(硬盘镜像)时，则可以避免服务的终止。当一个硬盘损坏时，整个硬盘系统将继续工作，数据不会丢失。同时对损坏硬盘的更换可以在计划中的系统维护时完成，如果使用热插拔硬盘，则可以在线更换。这种情况下的损坏是不会造成影响，同时修复也是在计划中进行。

第三种情况是选用一个采用热插拔硬盘的磁盘阵列系统，这样情况下的系统硬盘损坏，将不会造成任何损失。比如 RAID 采用的是 RAID5+Hot Spare，则在硬盘损坏时，磁盘阵列系统将继续工作，因为采用了磁盘冗余技术，只需完成的是在线更换损坏的硬盘即可，这时系统又恢复到出错前的状态。

1.4.4造成非计划宕机的原因

以下列举了常见的宕机原因：

A 硬件故障

B 文件系统溢出错误

C 运行在内存中的操作系统核心表溢出错

D 硬盘满

E 电压不稳、跳电

F 电源损坏

G 网络故障

H 软件漏洞（Bug）

I 应用出错

J 硬件设备 FirmWare 出错

K 灾难(火灾、水灾等)

L 管理员操作失误

非计划的系统服务终止，将会比正常宕机造成更多意料不可及的严重损失。

1.4.5非计划宕机的损害

非计划系统宕机将会导致非常严重的后果，比如 机场的航班导航系统故障，所有飞机将无法正常起降；医院的电脑系统出错，将导致患者无法结账、医生无法得到患者信息，甚至无法进行手术，总之，系统宕机所造成的损害将非常大。有些环境下，系统停止服务将导致事物处理无法进行，必将导致客户满意度的下降。

一个权威组织对系统非正常停止原因进行分析得到的结果是：

40%的异常终止是由于应用系统(Application)

20%的异常终止是由于硬件故障

另外的 40%是由于操作失误造成的

很多人仅在考虑到硬件故障时，会想到高可用需求，但从上面的分析，当构建一个高可用系统时，其它各种原因都该考虑在内。定义高可用需求时必须将软件、人的因素考虑在里面，当然其它的外部因素，像供电、气候、通信等也是考虑的因素。

高可用系统设置是为了减少或避免意外的系统故障，我们同样应该注意的是可能发生的不同类型的错误会导致系统的不同反应，并非所有的宕机都是由出错造成的，但出错肯定会造成损失。

1.5 高可用准备

实际上，实现高可用的主要障碍不是硬件，也不是软件出错，而是缺少一个好的流程和规范。通常概念中，高可用在有些人们眼中可能仅是一个概念，就像是一个技术而已，同时大家会认为高可用环境会增加系统故障点，实际上这些看法是片面的。我们知道，高可用的实现，需要一个非常好的软件来支持。

1.5.1定义高可用目标

准备一个高可用系统，首先要有一个明确的目标，服务级别定义（Service Level Agreement）指出了企业中用户对系统提供服务的需求是确定系统高可用级别的基础。SLA 可以明确常规的系统操作，列出了计划宕机时间（Planned downtime）和相应的性能方面需求。

下面列出几个常见的 SLA 项目：

A 系统在 24X5X52 环境下应该有 99.5%可用性；

B 在提供 Internet 服务情况下的相应时间应该在 1-2 秒，除了在进行备份时；

C 每周的一次系统完全备份（设计的维护时间段），并且在 90 分钟内完成；

D 每天需要完成增量备份，且在 30 分钟内完成；

E 系统出错的恢复时间应该在 15 分钟内；

SLA 可以认为是系统和用户之间的可用性需求约定。一个非常明确的定义可以使得提供相应服务的系统的硬件和软件需求变得十分清楚，同时使得根据性价比的情况选择合适的高可用方案。

注意：大型数据库服务器系统同一个提供 www 服务的多服务器系统的结构完全不同，数据库系统要求数据存储

共享，而提供 www 服务的系统则是不同的服务器提供同样的 www 访问服务，更多的倾向于 IP 访问的负载均衡。

1.5.2建立相应的高可用物理环境

实现高可用需要关注物理数据处理环境。由于非常小的损耗都是不可接受的，因此物理设备的安全性十分重要，配置物理环境时，需要考虑的包括系统的过载、连线的可靠、资源争用等。同时，高可用系统中的物理设备安全十分重要，应该加锁，比如热插拔硬盘的锁、磁盘阵列的安全密码设置等，防止非授权人员的非法进入。

1.5.3建立自动的处理流程

人为操作失误造成损失是很难控制的，因此在高可用系统中，建立尽量多的自动化流程十分重要，下面是应该是通过脚本自动完成的操作：

A 日常备份

B 日常维护处理

C 软件升级

D 部分错误的自动恢复

针对不同自动操作的实际脚本可能不同，但是通过自动的脚本操作，可以保证在系统出现故障时，能够最快的恢复服务。通常，恢复操作应该自动完成，以使系统故障时，能最快的恢复服务。

另外，高可用系统中的进程监控也是十分重要，进程的监控可以保证系统应用层面的进程在出现故障时，可以首先向高可用软件发出信号，以自动完成相应的操作，比如服务的重起、切换等，同时将完成操作及错误信息记录在日志文件中。因此，设计一个实现这样功能的高可用软件十分必要，可以降低发生错误的概率。

有些软件工具也能够对设备、资源进行监控、预警，但他们不同于高可用软件，这也是我们开发 LanderCluster 软件包的出发点之一。

1.5.4开发及测试环境

当设计、开发一套高可用软件时，软件的测试非常重要，必须在各种临时环境中完成各种可能的测试，否则，高可用可能变成更加低效的方案。一个高可用系统由多个部分组成，硬件环境、操作系统、数据库、应用程序、高可用软件包，他们工作在一起的时候，需要非常严格的测试。

测试可以分模块的进行，比如对网络连接的测试、进程管理的测试、文件系统（卷）的测试、IP 地址漂移的测试等，在手册的附件部分，提供了 LanderCluster 高可用环境的测试流程表,是十分专业的一套流程。

1.5.5按照需求配置硬件

高可用环境中，硬件是要求冗余的，包括两台服务器（我们称这种环境为双机容错），而部署多机高可用集群环境时，服务器则是多台，每台服务器拥有足够的 I/O 能力、内存容量、系统硬盘空间、网卡部件,使得配置能够在使用中符合系统的需求，并能够最大限度的降低系统停机时间。

1.5.6定义操作流程

当系统故障出现时，系统管理员和操作员必须知道以下情况的操作：

A 何时自动恢复应该开始；

B 何时系统故障需要管理员或操作员来操作；

C 何种情况需要支持电话；

D 何时需要进行灾难恢复；

1.5.7灾难恢复准备

用户考虑高可用的最终目的是有一个清楚的操作流程，除了保证系统可以不间断的提供优质服务，还有就是在自然和人为原因导致系统灾难发生时的对策。在系统灾难发生后，应该有一个经过严格测试的恢复流程作为系统的灾难恢复准备。

灾难备份和保障系统，将是另外一个需要讨论的安全概念，不是处在集群概念中的，因此可以在联鼎的应用容灾产品文档中获得相关介绍。

1.5.8管理员的培训

系统管理员必须经过严格的培训，才可以得心应手的管

理好一个高可用系统，因为高可用环境完全不同于传统的硬件环境。管理员必须对高可用有深刻的认识，同时必须有完整的培训流程来让他们掌握针对各种不同的故障的操作流程。例如 RAID 损坏、文件系统损坏、网络失效等情况下的实际判断和操作，哪些问题可以自己处理，哪些问题需要寻求技术支持单位的帮助。当然，管理员可能由于大多数精力放在业务上，而且不可能面面俱到，关键是尽量减少错误的操作。

1.5.9记录每一个细节

文档资料的重要性是不言而喻的。专业的系统安装和维护，都应该具有一个完整的事件记录。一般的系统安装、设置在完成以后，没有留下记录，导致后续维护的困难，增加了系统故障的概率。因此，养成一个好的习惯，详细记录每个系统的配置变动信息、系统安装设置信息、故障及处理过程信息相当的重要。联鼎软件作为专业的技术服务商，他们的技术服务流程就是很值得借鉴，无论是安装设置、顾问支持还是现场故障处理，他们都有非常丰富的文档资料留给用户，一方面体现了其专业的服务流程，另外也为用户留下了珍贵的技术文档。对管理员来说应该养成这样好的习惯，将会受益匪浅。

1.5.10高可用系统实现的要点

高可用系统是建立在可靠性组成的最顶层。许多企业级系统部件环境具有以下特征，是实现高可用环境的基本要求。

A　硬件环境的高可靠

B　软件的品质保证

C　智能的系统诊断

D　强大的系统管理工具集

E　维护和售后支持

高可用系统不会因为这些特性而难于实现，但这些特性确实促进了集群技术的发展。

1.5.11底层硬件的可靠性

我们知道，高可用环境通常包括两台（多台）服务器及存储子系统（通常为磁盘阵列系统），这些基本硬件的可靠性是高可用系统的最基本要求。我们可以想象，一个双机系统中采用了不可靠的 RAID 系统，那么，再可靠的服务器也是惘然；同样采用了非常可靠的 RAID 系统（99.999%），而两台服务器的可靠性又很低，那么整个高可用的可靠性也要打折扣。因此在实现一个高可用系统时，底层硬件设备的选型和配置十分重要。

1.5.12软件的品质

从操作系统到数据库软件、到应用程序，软件的品质同样影响高可用系统的可靠，比如操作系统可能采用 Microsoft Windows 、Linux、SCO UnixWare 等，它们有不同的可靠性和安全性；运行的数据库可能为 Oracle、Sybase、Informix，或 SQL Server 等，它们不一定可以运行在所有平台，而且在不同平台的稳定性不同。这些都是我们要考虑的问题，定义一个高可用系统并不是一件很简单的事。

1.5.13集群技术服务

从前面的论述中，可以知道高可用系统不是一个很简单的环境，实际上它的正常运行，需要经过合理配置的系统软件、硬件协同提供高可用的服务。那么，对于这个环境，能够持续的提供包括现场服务十分重要。联鼎软件（Lander Software）是专门从事数据安全技术服务的厂商，其自行研发的高可用产品 LanderCluster 在众多硬件、软件环境平台经过非常严格的测试，和拥有大量的行业用户，证明 LanderCluster 软件包是真正经得起考验的高可用产品。联鼎产品用户得到的不单单是一个软件产品、一个方案，他们更是得到了联鼎技术团队提供的持续的技术服务，从而更增强了系统的可靠性。具体内容参考附件部分（联鼎双机服务协议）。

前面的描述，使我们知道了高可用环境的概念、技术、实现等，这样，您就可以为自己的系统规划一个高可用环境了 。

1.6　集群避免单点故障

如何达到高可用？集群（通过网络、共享存储组成的计算机集合），是常见的方式，通常的集群是运行 HP-UX、Linux、SCO Unix、

MS Windows Server 等操作系统松散结合的主机系统，

通常的高可用环境，包括我们 LanderCluster 支持的运行环境概念上可以看成是经过裁剪的集群环境 。

我们这里主要讨论如何配置合适的高可靠性的软件和硬件，并且做到硬件冗余,避免单点故障的发生。

需要了解的包括以下几个部分：

A 如何确定单点故障

B 避免电源的单点故障

C 避免硬盘的单点故障

D 避免系统处理单元（SPU）的单点故障

E 避免网络的单点故障

F 避免软件的单点故障

G 配置高可用集群系统

并非所有的系统都能够做到无任何单点故障，这里我们探讨的是系统配置上尽量做到无单点故障，然后通过一个好的高可用软件实现整个系统高可用。

1.6.1单点故障

一个非常可靠的独立运行的服务器同样会有很多的单点故障，单点故障（Single Point Of Failure---SPOF） 指系统的软件或硬件在其失去功能时，会导致整个系统停止服务，则该点称之为单点故障。通常，一个没有备份或冗余、单独运行的的部件都是单点故障。

例如在下面描述的一个典型环境，环境如下：某个企业的业务系统运行在一台 Intel 构架的 PC Server 上，操作系统是 RedHat Linux,这是一个典型的 Client/Server 结构的环境，服务器从客户端取得业务数据，经过处理存储在主机的硬盘上，并返回部分结果。这样的一个单独运行的服务器会有哪些问题呢？

A CPU 的损坏会导致整个系统停止工作；

B 网络连接的失效会导致客户端无法得到主机的服务；

C 系统由于故障重启，应用进程或服务进程工作在非正常模式下时，客户端无法得到服务；

D 系统盘的损坏会导致系统宕机；

E 数据盘的损坏会导致应用的无法运行；

F 系统电源失效会导致系统宕机，并数据丢失；

G 硬盘空间不够、交换区满会导致系统宕机；

如何避免单点故障列表

系统部件	部件损坏的结果	如何避免该单点故障
单主机	系统停止工作	提供备份服务器，通常用高可用性集群实现可切换
单网卡	客户端无法连接	配置冗余网卡（有些系统支持）
单网络	客户端无法连接	配置冗余子网
单系统盘	系统宕机（操作系统无法启动）	镜像系统硬盘（Mirror）
单数据盘	数据库数据丢失	RAID 或镜像
单电源	服务器无法工作	配置冗余电源
单硬盘控制卡	硬盘无法访问，系统停止服务	配置冗余控制卡
单操作系统	服务丢失	提供错误切换功能，或恢复功能
应用程序	服务丢失	提供应用自动重启或应用错误恢复功能
人为错误	服务丢失，直到恢复	尽量减少人为操作，详细规定好系统管理规范

1.6.2避免电源的单点故障

很简单，强烈建议在配置服务器和存储设备时，必须要求是双电源供电，同时必须有两路不同的电源供给，或必须插在不同的电源插座上。有些电源单路输入，内部两个电源模块是不能称为标准双电源的。同时建议使用 UPS 进行供电，减少由电源引起的出错概率。

1.6.3避免硬盘的单点故障

好的高可用系统中的主机系统盘要求配置成 RAID 1（镜像盘），保证系统盘无单点故障，因为如果系统盘损坏，会导致系统重建和大量的恢复操作。如果是数据盘损坏，则系统本身可以正常运行，但是应用程序无法运行，只有在数据盘更换，从备份恢复好原来的数据后，才能使用。无论系统盘损坏还是数据盘损坏，都会造成一定的损失。因此，冗余配置是十分必要的。通常有两个方法来解决这个问题，磁盘阵列或软件方案，两种方法各有长处。

1.6.4通过磁盘阵列（RAID)保护数据安全

RAID 是“独立磁盘冗余阵列”（Redundant Array of Independent Disks）（最初为“廉价磁盘冗余阵列”）的缩略语，1987 年由 Patterson，Gibson 和 Katz 在加州大学伯克利分院的一篇文章中定义。RAID 阵列技术允许将一系列磁盘分组，以实现提高可用性的目的，并提供为实现数据保护而必需的数据冗余，有时还有改善性能的作用。随着计算机技术的快速发展，RAID 已经从高端服务器市场日益步入寻常百姓家。

RAID 级别可以通过软件或硬件实现。许多但不是全部网络操作系统支持的 RAID 级别至少要达到 5 级，RAID 10、30 和 50 只有在磁盘阵列控制器控制下才能实现。基于软件的 RAID 需要使用主机 CPU 周期和系统内存，从而增加了系统开销，直接影响系统的性能。磁盘阵列控制器把 RAID 的计算和操纵工作由软件移到了专门的硬件上，一般比软件实现 RAID 的系统性能要好。

有三个因素将影响您对 RAID 级别的选择：可用性（数据冗余），性能和成本。如果不需要可用性，那么 RAID-0 将带来最佳性能。如果可用性和性能很重要而价格并不重要，那么选择 RAID-1 或 RAID-10（视磁盘数而定）。如果价格、可用性和性能同样重要，那么选择 RAID-3，RAID-30，RAID-5 或 RAID-50。

RAID 技术比较：

表 1	RAID 0	RAID 1	RAID 3	RAID 5
名称	条带	镜像	专用校验条带	校验条带分散
允许故障	否	是	是	是
冗余类型	无	副本	校验	校验
热备用操作	不可	可以	可以	可以
硬盘数量	一个以上	两个	三个以上	三个以上
可用容量	最大	最小	中间	中间
减少容量	无	50%	一个磁盘	一个磁盘
读性能	高(盘数决定)	中间	高	高
随机写性能	最高	中间	最低	低
连续写性能	最高	中间	低	最低
典型应用	无故障 迅速读写	允许故障 小文件、随机数据写入	允许故障 大文件连续数据传输	允许故障 小文件随机数据传输

表 2	RAID 10	RAID 30	RAID 50
名称	跨越镜像 阵列	跨越专用 校验阵列	跨越分散 校验阵列
允许故障	是	是	是
冗余类型	副本	校验	校验
热备用操作	可以	可以	可以
磁盘数量	跨越 2 个阵列 4,6,8,10,12,14 或 16 6,8,10,12,14 或 16	跨越 3 个阵列 6 9,12 或 15 9,12 或 15	跨越 4 个阵列 8 12 或 16 12 或 16
可用容量	最小	中间	中间
减少容量	50%	每个阵列中一个磁盘	每个阵列中一个磁盘
读性能	中间	高	高
随机写性能	中间	最低	低
连续写性能	低	中间	最低
典型应用	允许故障高 速度小文件随机数据写入	允许故障高 速度大文件、连续数据传输	允许故障高 速度小文件、随机数据传输

使用 RAID 保护数据的好处是：

A　可以在线更换出错的硬盘；

B　可以设置 Hot Spare 硬盘，使在某块硬盘损坏时，自动替换出错的硬盘；

C　可以实现海量存储（达到 Tb 级）；

D　灵活的配置，配置成不同级别的 RAID;

E　在某些环境下性能极好；

F　通常的配置是双电源、风扇，如果配置双控制卡，可以有效的避免单点故障；

1.7　认识 SAN 存储结构

1.7.1存储区域网介绍

SAN（Storage Area Network）是指存储局域网，通过特殊的网络来使网络中的存储设备可以让该网络中的不同服务器直接访问，通常是通过光纤结构实现互联。随着存储技术的不断发展，SAN 结构的存储系统将成为主流，因为相对传统的存储结构，它具有众多的优点。通常，SAN 结构的系统必须要配置成高可用，或者说服务器必需可以能够通过可靠的网络连接访问存储。SAN 结构中的不同部件，如控制器、磁盘阵列等必需能够冗余。

当前，我们正处在一个信息爆炸的时代，数据的存储量已经不仅仅是用 KB、MB、GB 甚至 TB 来计算，在不远的将来，人们所谈论的将是 PB（1petabyte=1,000terabytes）甚至 EB（1exabyte=1,000petabytes）。根据 IDC 公司的统计报告，企业数据的增长速度是每年 100%。在企业的作业系统和数据采掘中，大量的、频繁的数据移动将会对用户的区域网或者广域网造成巨大的影响。此外，如何使分布的存储设备（存储农场，Storage Farm）更加有效的运行，也是摆在每个用户的问题。

计算机的发展历史来看，从最早的服务器 /客户机（Server/Clinet）模式，到今天的网络计算（Network Computing）环境，今后的移动计算（Pervasive Computing）环境，对数据的请求不再受时间和空间的限制。随之而来的问题是，当前的数据多分布在与服务器相连的独立存储之上，从而造成所谓的“信息孤岛”（Information Island）的现象。这使数据的存储、利用、分析和管理都非常地复杂。

采用存储区域网（SAN），可以通过快速的、专用的光纤网络，将上百个甚至几千个存储设备连接起来，组成低成本的、易于管理的存储区域网络。存储区域网不仅可以减少数据移动对现有的网络系统的压力，从而降低存储的成本，而且可以通过将存储设备的集中，方便地进行监视和调整，从而实现灵活方便的管理。

对于用户，由于数据量非常大 （包括数据库、音像、图像和文字信息），而处理平台又是多种多样，采用 SAN 架构，可以实现存储设备的跨平台使用，存储设备的统一管理，存储容量的动态分配，提高存储设备的利用率，减少维护和使用的成本。同时，可以实现数据的集中存储和访问，为今后应用的整合打下基础。

采用存储区域局（SAN），数据的备份可以不再通过网络，大大减少了 LAN 的压力，从而称为与本地局域网无关（LAN Free）以及与服务器无关（Server Less）)的备份。

SAN 提供了一种与现有 LAN 连接的简易方法，并且通过同一物理通道支持广泛使用的 SCSI 和 IP 协议。SAN 不受现今主流的、基于 SCSI 存储结构的布局限制。特别重要的是，随着存储容量的爆炸性增长，SAN 允许企业独立地增加它们的存储容量。 SAN 方案也使得管理及集中控制的实现简化，特别是对于全部存储设备都集群在一起的时候。而且，光纤接口提供了 10 公里的连接长度（通过特殊网络设备，距离可达 70 公里，甚至更远），这使得实现物理上分离的、不同机房的存储变得容易。

在传统的基于 SCSI 技术的存储方式中，磁盘上的数据是服务器的专有资源，存储任务依赖于服务器及其所挂接的 LAN。由于这种技术本身的局限性以及存储任务对网络带宽的消耗越来越多，并行 SCSI 技术已渐渐不能够满足客户存储的需求。而 SAN 的所推出首先使服务器同存储阵列之间的连接方式发生了根本性的变革，基于 Fibre Channel（同时具备网络和通道特性，能够以千兆位速度进行数据传输的技术）的 SAN 改变了传统服务器与磁盘阵列的主从关系。位于 SAN 上所有设备均处于平等的地位，任何一台服务器均可存取网络上任何一台存储设备，通过 Fibre Channel 高带宽和强大的 IO 处理能力，SAN 技术在可连接性、可扩展性以及性能方面解决了 SCSI 技术无法解决的问题，成为存储领域具有强大生命力的新技术。

另外，在具体存储应用方式上，对于一个 IT 机构或部门

来讲，传统的存储是在 LAN 上进行，大大占用了 LAN 的带宽资源，另外由于这种传统的存储方式多是在以太网上以 TCP/IP 协议传输数据，在层层打包之后资源会有较大的开销。由于在 SAN 存储网络设计方式中，文件传输是直接在界面卡（SCSI、SSA、FC）与存储设备进行交互，无须经过网络传输，此时原先数据传输所占用的网络带宽，可大大被释放。SAN 也支持 IP 协议，但由于其针对存储数据传输的特点进行设计，当需要有大量或者大块的数据在 SAN 上进行传输时，基于 Fibre Channel 的传输技术更有优势，当客户端在 LAN 上请求来自服务器的数据时，服务器将在 SAN 上的存储设备中检索数据，由于这种方式对数据的处理没有 IP 打包方面的开销，所以能够更有效的提交数据。

存储区域网是有管理的、高速的存储网络，它包含了存储企业商务信息的多供应商的存储系统、存储管理软件、应用服务器和网络硬件。简单来说， SAN 是从主机应用平台分离出来的作为一个整体来管理的一系列硬件和软件。

SAN 存储区域网的基本内容包括管理、使用和服务，它将光纤通道集线器（Hub），交换机（Switch）、导向器（Director）等硬件与软件管理功能结合为一体，各种设备和软件不论是否出自 IBM 公司都可以密切配合，随时随地的实现信息的存储、访问、共享和保护。凭借在 IT 系统的规划、设计和实施的丰富经验，IBM 将会为端到端的 SAN 解决方案提供完善的支持、服务和培训。

1.7.2 SAN 的优势

SAN 的价值在于增加了存储系统的可用性、灵活性以至于减少了应用系统的宕机时间。

以下我们给出 SAN 的优势：

存储系统集中

在 SAN 未出现以前，在计算中心的公用区域内将设备进行物理集中通常是不可能的，若有可能的话，却又要求昂贵和专有的扩展技术。通过在存储资源和服务器之间引入网络，解决了这样一个难题。

增加容量

当所有的设备都与 SAN 相连，那么为一个或多个服务器增加存储容量就变得非常简单。集中化的磁盘存储系统容许多台服务器使用由 SAN 连接的磁盘存储设备组成的公用存储池。 此时可以在一个磁盘子系统内或跨多个磁盘阵列子系统对磁盘存储资源进行汇集，同时将汇集的磁盘容量指定给由服务器操作系统支持的独立文件系统。其中，服务器可以是异构的 UNIX、Windows、Linux 等。

存储设备可以动态地增加到磁盘池内，并且根据需要随时分配给与 SAN 相连的服务器使用。由于存储设备做到了与服务器直接相连，并且存储容量的整合实现了容量的有效扩展，所以，与独立文件服务器的间接连接相比，这种磁盘汇集方法实现了有效的磁盘资源共享。

磁带集中解决了当今开放系统环境中面临的问题，在那里，多台服务器不能跨多台主机共享磁带资源。目前在主机之间进行设备共享的方法是：人工将磁带设备从一台主机切换到另一台主机；或者是利用分布式编程来编写服务器之间进行通信的应用程序。

服务器群集

由于异构服务器的群集可以将数据当作是一个单一的系统映像来观看，所以 SAN 结构实际上是以一种全共享的方式提供给可扩展的群集。虽然现在使用多路径的 SCSI 使这种想法成为可能，但可扩展性 仍然是存在的一个问题，因为 SCSI 的距离受到限制。一般的 SCSI 允许传送距离 25 米，同时 SCSI 连接器的尺寸也限制了连接到服务器或子系统上的连接数量。

SAN 允许在分布式处理应用环境中进行有效的负载均衡，在一台服务器上受到处理器限制的应用可 以在多台服务器上利用更大的处理器能力得到执行。为了做到这一点，服务器必须能访问相同的数据卷，同时应用程序或操作系统必须提供对数据访问的串行化服务。

除了这些优势外， SAN 结构还可以进行开发用于故障恢复，当主系统出现故障时，辅助系统接管主系统的工作，并直接访问主系统使用的存储设备。这样就消除了由于处理器失效而造成的停机现象，从而使群集系统环境下的可靠性大大提高。

建立一个基于 SAN 的数据存储与备份系统不仅可以提高系统的可靠性，同时还能提高系统的性能，随着系统的不断扩展，存储系统的性能和容量都可以灵活的扩展。

存储概念的再塑

数据存储（Storage）在传统意义中是磁带、内存、硬盘、磁盘、CD 等一些列介质和设备。但数据存储技术和合理应用及系统架构应该如何系统考虑？一直没有一个很有针对性的理论。数据存储基本是被动地跟着主机系统的建设而设立，其主要功能是满足某种运算。随着各种应用的不断发展，存储设备的性质也在慢慢的变化着。从

一种为计算服务的辅助设备变为了受到多重保护的核心设备，其主要原因是它所存储的数据拥有巨大的价值。同时，不断发展的需求也推动着存储技术的不断向前发展。

1.7.3 SAN 的适用环境

用户需要快速网络、使用电子商务及 OLTP 应用系统的单位、需要整合多种服务器及许多存储设备的单位、简化现行存储系统管理。SAN（存域网）是指独立于异构计算网络系统之外的几乎拥有无限存储容量的高速网络。其采用高速的光纤通道作为传输媒介，以 FCP/SCSI 协议作为存储访问协议，将存储子系统网络化、开放化、虚拟化、智能化，实现真正的高速、安全、共享存储。

1.7.4 存储区域网（SAN）的特点

提升存储系统资源使用效率及平衡工作量

持续且具弹性的存储系统运作 （Seamless scaling of storage operations）

改善数据存取速度 （Improved data access with reduced latency）

更有效的数据保护作业 （data protection）

快速的数据分享 （File sharing of a LAN at the speed of a SAN）

集中管理，降低总运作成本

业务数据恢复 （Backup/restore）

存储系统集成 （Storage consolidation ,disk and tape pooling/sharing）

灾难容错 （Disaster Recovery）

高可靠性 （High availability / clustering）

1.7.5SAN 存储系统五大组成

Server （服务器）

Storage（存储设备）

SAN fabric（连接设备）

Software（管理软件）

Service（服务）

1.8 认识 ISCSI 存储结构

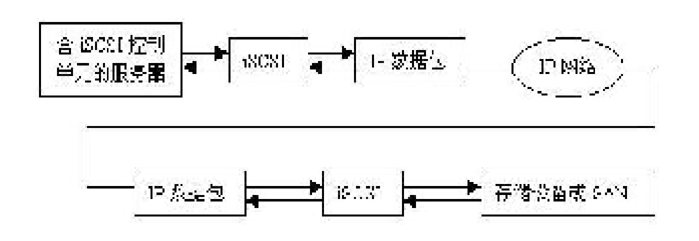
1.8.1ISCSI 技术简介

此技术于 2003 年 2 月 11 日，由 IETF(Internet Engineering Task Force，互联网工程任务组)正式通过;它由 IBM、思科共同发起，是一种基于网络的数据存储技术，具有硬件成本低廉，操作简单，扩充性强，传输速度快等特点。

ISCSI(互联网小型计算机系统接口)是一种在 Internet 协议网络上，特别是以太网上进行数据块传输的标准。它是由 Cisco 和 IBM 两家发起的，并且得到了 IP 存储技术拥护者的大力支持。是一个供硬件设备使用的可以在 IP 协议上层运行的 SCSI 指令集。简单地说，ISCSI 可以实现在 IP 网络上运行 SCSI 协议，使其能够在诸如高速千兆以太网上进行路由选择。

1.8.2ISCSI 工作流程

当客户端发出一个数据、文件或应用程序的请求后，操作系统机会根据客户端请求的内容生成一个 SCSI 命令和数据请求，SCSI 命令和数据请求通过封装后会加上一个信息包标题，并通过以太网传输到接收端;当接收端接收到这个信息包后，会对信息包进行解包，分离出的 SCSI 命令与数据，而分离出来的 SCSI 命令和数据将会传输给存储设备，当完成一次上述流程后，数据又会被返回到客户端，以响应客户端 ISCSI 的请求。其流程大致可参看如图所示。



1.8.3技术优势

在本文开头，其实已经解释了此技术的主要优势，及硬

件成本低廉，操作简单，扩充性强，传输速度快等。

硬件成本低廉:

基于 ISCSI 技术的适配卡、交换机和缆线这些产品的价格相对较低，而且 ISCSI 可以在现有的网络上直接安装，并不需要更改企业的网络体系，这样可以最大程序的节约投入。

操作简单:

此技术主要是通过 IP 网络实现存储资源共享，只需要现有的网络功能即可管理，其设置也非常简单。

扩充性强:

由于 ISCSI 存储系统可以直接在现有的网络系统中进行组建，并不需要改变网络体系，加上运用交换机来连接存储设备，对于需要增加存储空间的企业用户来说，只需要增加存储设备就可完全满足，因此，ISCSI 存储系统的可扩展性高。

传输速度快:

由于 ISCSI 的数据传输速度是根据以太网络速度而变化的，因此当以太网速度增加时，ISCSI 的数据传输速度也将相应的加快。而且其传输过程由于基于网络进行，因此没有范围限制。

1.9 64 位技术

64 位技术是 64 位技术是相对于 32 位而言的，这个位数指的是 CPU GPRs（General-Purpose Registers，通用寄存器）的数据宽度为 64 位，64 位指令集就是运行 64 位数据的指令，也就是说处理器一次可以运行 64bit 数据。64bit 处理器并非现在才有的，在高端的 RISC（Reduced Instruction Set Computing，精简指令集计算机）很早就有 64bit 处理器了，比如 SUN 公司的 UltraSparc Ⅲ、IBM 公司的 POWER5、HP 公司的 Alpha 等。

64bit 计算主要有两大优点：可以进行更大范围的整数运算；可以支持更大的内存。不能因为数字上的变化，而简单的认为 64bit 处理器的性能是 32bit 处理器性能的两倍。实际上在 32bit 应用下，32bit 处理器的性能甚至会更强，即使是 64bit 处理器，目前情况下也是在 32bit 应用下性能更强。所以要认清 64bit 处理器的优势，但不可迷信 64bit。

要实现真正意义上的 64 位计算，光有 64 位的处理器是不行的，还必须得有 64 位的操作系统以及 64 位的应用软件才行，三者缺一不可，缺少其中任何一种要素都是无法实现 64 位计算的。

1.10 虚拟化

虚拟化是一个广义的术语，在计算机方面通常是指计算元件在虚拟的基础上而不是真实的基础上运行。虚拟化技术可以扩大硬件的容量，简化软件的重新配置过程。CPU 的虚拟化技术可以单 CPU 模拟多 CPU 并行，允许一个平台同时运行多个操作系统，并且应用程序都可以在相互独立的空间内运行而互不影响，从而显著提高计算机的工作效率。

虚拟化技术与多任务以及超线程技术是完全不同的。多任务是指在一个操作系统中多个程序同时并行运行，而在虚拟化技术中，则可以同时运行多个操作系统，而且每一个操作系统中都有多个程序运行，每一个操作系统都运行在一个虚拟的 CPU 或者是虚拟主机上；而超线程技术只是单 CPU 模拟双 CPU 来平衡程序运行性能，这两个模拟出来的 CPU 是不能分离的，只能协同工作。

1.10.1纯软件虚拟化

纯软件虚拟化解决方案存在很多限制。“客户”操作系统很多情况下是通过 VMM(Virtual Machine Monitor，虚拟机监视器)来与硬件进行通信，由 VMM 来决定其对系统上所有虚拟机的访问。(注意，大多数处理器和内存访问独立于 VMM，只在发生特定事件时才会涉及 VMM，如页面错误。)在纯软件虚拟化解决方案中，VMM 在软件套件中的位置是传统意义上操作系统所处的位置，而操作系统的位置是传统意义上应用程序所处的位置。这一额外的通信层需要进行二进制转换，以通过提供到物理资源(如处理器、内存、存储、显卡和网卡等)的接口，模拟硬件环境。这种转换必然会增加系统的复杂性。此外，客户操作系统的支持受到虚拟机环境的能力限制，这会阻碍特定技术的部署，如 64 位客户操作系统。在纯软件解决方案中，软件堆栈增加的复杂性意味着，这些环境难于管理，因而会加大确保系统可靠性和安全性的困难。

1.10.2CPU 虚拟化技术

CPU 的虚拟化技术是一种硬件方案，支持虚拟技术的 CPU 带有特别优化过的指令集来控制虚拟过程，通过这些指令集，VMM(Virtual Machine Monitor，虚拟机监视器)会很容易提高性能，相比软件的虚拟实现方式会很大程度上提高性能。虚拟化技术可提供基于芯片的功能，借助兼容 VMM 软件能够改进纯软件解决方案。由于虚拟化硬件可提供全新的架构，支持操作系统直接在上面运行，从而无需进行二进制转换，减少了相关的性能开销，极大简化了 VMM 设计，进而使 VMM 能够按通用标准进行编写，性能更加强大。另外，在纯软件 VMM 中，目前缺少对 64 位客户操作系统的支持，而随着 64 位处理器的不断普及，这一严重缺点也日益突出。而 CPU 的虚拟化技术除支持广泛的传统操作系统之外，还支持 64 位客户操作系统。

1.10.3文件虚拟化

文件虚拟化(File Virtualization)是在文件服务器和访问这些文件服务器的客户机之间创建一个抽象层。一旦应用，文件虚拟化层管理跨服务器的文件和文件系统，允许管理员向客户机提供一个所有服务器的逻辑文件挂接。这台服务器继续托管文件数据和元数据。

虽然这种安排好想象不必要地增加了 IT 开销，但是，文件虚拟化提供了一些关键的优势，包括一个全局命名空间用来给网络文件服务器上的文件加索引。此外，这种虚拟文件存储整合允许文件服务器之间共享访问存储容量。文件服务器之间实施的数据迁移对于最终用户和应用程序都是透明的。这在分层次的存储基础设施中是理想的。简言之，文件虚拟化允许企业访问网络文件服务器上隔离的存储容量并且在上面进行无缝的文件迁移。

文件虚拟化可以部署为一台设备或者一台运行文件虚拟化软件的现成的服务器。这种选择基本上是根据成本以及有关的管理和破坏水平确定的。最常用的部署选择是设备。这种设备有四种不同的架构：带外、带内、这两者的结合和分离路径(Split-Path)。

并为所有的文件虚拟化部署从长远看是成功的。有些机构也许会退回(撤销)他们的部署。这对于文件服务器和网络附加存储平台来说是一个破坏性非常大的过程。在极端的情况下，退回可能需要机构卸载数据、删除文件虚拟化层，然后重新格式化和重新装载全部数据。经销商通常能够帮助识别潜在的退回问题，提供减轻破坏的建议。用户在一般部署之前通常要测试其退回的程序。

文件虚拟化受到可伸缩性的限制。可伸缩性包括文件系统、文件、服务器或者输入/输出性能。文件虚拟化平台还必须要兼容当前的基础设施。这样，它就能够与现有的存储系统和交换机一起工作。要防止出现潜在的问题，文件虚拟化平台应该经常进行适当的可伸缩性和兼容性测试。

1.10.4存储虚拟化

简单的讲，虚拟存储(Storage Virtualization)，就是把多个存储介质模块(如硬盘、RAID)通过一定的手段集中管理起来，所有的存储模块在一个存储池中得到统一管理。这种可以将多种、多个存储设备统一管理起来，为使用者提供大容量、高数据传输性能的存储系统，就称之为虚拟存储。

存储虚拟化的基本概念是将实际的物理存储实体与存储的逻辑表示分离开来，应用服务器只与分配给它们的逻辑卷(或称虚卷)打交道，而不用关心其数据是在哪个物理存储实体上。

逻辑卷与物理实体之间的映射关系，是由安装在应用服务器上的卷管理软件(称为主机级的虚拟化)，或存储子系统的控制器(称为存储子系统级的虚拟化)，或加入存储网络 SAN 的专用装置(称为网络级的虚拟化)来照管的。

主机级和存储子系统级的虚拟化都是早期的、比较低级的虚拟化，因为它们不能将多个，甚至是异构的存储子系统整合成一个或多个存储池，并在其上建立逻辑虚卷，以达到充分利用存储容量、集中管理存储、降低存储成本的目的。

只有网络级的虚拟化，才是真正意义上的存储虚拟化。它能将存储网络上的各种品牌的存储子系统整合成一个或多个可以集中管理的存储池(存储池可跨多个存储子系统)，并在存储池中按需要建立一个或多个不同大小的虚卷，并将这些虚卷按一定的读写授权分配给存储网络上的各种应用服务器。这样就达到了充分利用存储容量、集中管理存储、降低存储成本的目的。

目前存储虚拟化的发展尚无统一标准，从存储虚拟化的拓扑结构来讲主要有两种方式:即对称式与非对称式。

对称式存储虚拟技术是指虚拟存储控制设备与存储软件系统、交换设备集成为一个整体，内嵌在网络数据传输路径中;非对称式存储虚拟技术是指虚拟存储控制设备独立于数据传输路径之外。

1.10.5虚拟主机

虚拟主机(Virtual Host/ Virtual Server):是使用特殊的软硬件技术，把一台真实的物理计算机主机分割成多个的逻辑存储单元，每个单元由于没有物理实体，但是每一个物理单元都能像真实的物理主机一样在网络上工作——独立的域名、IP 地址(或共享的 IP 地址)、完整的 Internet 服务器功能。

1.11 刀片服务器

刀片服务器是一种 HAHD（High Availability High Density，高可用高密度）的低成本服务器平台，是专门为特殊应用行业和高密度计算机环境设计的。其中每一块"刀片"实际上就是一块系统主板。它们可以通过本地硬盘启动自己的操作系统，如 Windows NT/2000、Linux、Solaris 等等，类似于一个个独立的服务器。在这种模式下，每一个主板运行自己的系统，服务于指定的不同用户群，相互之间没有关联。不过可以用系统软件将这些主板集合成一个服务器集群。在集群模式下，所有的主板可以连接起来提供高速的网络环境，可以共享资源，为相同的用户群服务。在集群中插入新的"刀片"，就可以提高整体性能。而由于每块"刀片"都是热插拔的，所以，系统可以轻松地进行替换，并且将维护时间减少到最小。值得一提的是，系统配置可以通过一套智能 KVM 和 9 个或 10 个带硬盘的 CPU 板来实现。CPU 可以配置成为不同的子系统。一个机架中的服务器可以通过新型的智能 KVM 转换板共享一套光驱、软驱、键盘、显示器和鼠标，以访问多台服务器，从而便于进行升级、维护和访问服务器上的文件。

1.12 避免系统处理单元单点故障

一个服务器系统的处理单元通常是由一组部件组成，它们通常都有可能发生故障。其中比较重要的几个部分是：

A CPU（一个或多个）

B I/O 控制卡（SCSI、I/O 板等）

C 内存

任何的这些部件的损坏，都将导致系统的拒绝服务。停止工作并更换新的配件，使得系统能够正常工作。如果一个系统不容许任何的出错，则只有两个办法：

A 所有部件冗余（通常所说的容错机）；

B 采用集群技术（实现高可用）

1.12.1单机高可用的实现

单机高可用通常指那些具有容错功能的服务器、专业容错机等，能实现这些功能的基本都是高端服务器。它们通常可以提供包括如下部件的硬件冗余：

A 处理器

B I/O 控制卡

C 网卡

D 内存部件（板或模块）

E 电源及连接线

这样的服务器包括 HP 的 SuperDome 和 Stratus 等，当然，它们也可能存在单点故障，比如系统的总线、系统时钟等，它们很难实现冗余，因此这能做到有限能力得高可用。

1.12.2通过集群来避免单点故障

高可用集群可以使您有效的避免处理单元（SPU）的单点故障问题。高可用集群环境可以大大的减少由于 SPU 损坏造成系统宕机的时间，可以容许用户在一个处理单元损坏时，进行修复而系统不会停止服务。在高可用集群系统中，一个或多个系统单元作为工作单元（Primary）的备份单元。这些备份单元可以是运行状态（Active）或是等待状态（Standby）。如果是运行的单元，在运行自身应用的同时，还作为主服务器的备份。而等待状态的单元在发生切换之前处于空闲状态（Idle），当然，Standby 的服务器也可以用于处理其它事物。高可用集群十分灵活，它们通常比容错的单机具有更好的性能价格比。

注意：SPU 和节点（Node）不同，节点是指集群中的一台服务器，而 SPU 指系统处理单元，包括一个或多个 CPU、内存及电源等部件组成的集合，节点包含 SPU。

集群中的节点之间通过网络连接，为客户端提供服务，

它们通过网络进行心跳侦测，了解对方的运行状态，当节点的 SPU 发生故障，备份节点将接管资源、启动服务，使得客户端可以持续的得到服务，这当中会有很短的中间状态，这样一个接管操作也称之为切换（Failover）。

切换工作的完成，需要特殊的软件来实现，我们的 LanderCluster 就是这样的高可用软件。不同的集群系统可能使用不同的技术，集群技术的顶级技术目前是 DEC（HP） 的 TrueCluster，可以通过内存通道实现节点间的内存镜像。还有其它象 HP 的 McServiceGuard、IBM 的 HACMP 等，它们是针对特定的操作系统平台的高可用方案，有着不同的特点。

高可用环境中的磁盘阵列系统物理上同两个节点分别连接，保证上面的数据两台服务器均可以访问，并且在一个节点故障时，另外的系统可以接管上面的文件系统/裸设备。每个节点均有自己的根盘（root disk），用于安装操作系统和应用程序。LanderCluster 系统可以支持多节点环境的高可用，多个节点通过 SCSI 或 FC 同共享存储设备连接，每个工作节点均可以访问存储设备上的文件系统。访问分为共享和独占，在某些高端集群系统环境下，需要数据访问共享（同时对文件进行读写），则要求软件具有锁功能，保证读写的一致性。如运行 ORACLE OPS 环境下，多个服务器同时运行在一个共享的 Database 上，这时必需通过软件提供的锁管理技术保证系统正常提供访问。

1.13 几种类型存储技术介绍

1.13.1DAS-直接连接存储

DAS（Direct Attached Storage–直接附加存储）是指将存储设备通过 SCSI 接口或光纤通道直接连接到一台计算机上。

DAS 的适用环境为：

1) 服务器在地理分布上很分散，通过 SAN 或 NAS 在它们之间进行互连非常困难时(商店或银行的分支便是一个典型的例子)；

2) 存储系统必须被直接连接到应用服务器（如 Microsoft Cluster Server 或某些数据库使用的“原始分区”）上时；

3) 包括许多数据库应用和应用服务器在内的应用，它们需要直接连接到存储器上，群件应用和一些邮件服务同样包括在内。

当服务器在地理上比较分散，很难通过远程连接进行互连时，直接连接存储是比较好的解决方案，甚至可能是唯一的解决方案。利用直接连接存储的另一个原因也可能是企业决定继续保留已有的传输速率并不很高的网络系统。

1.13.2NAS-网络连接存储

NAS 和 SAN 的出现响应了三种重要的发展趋势：网络正成为主要的信息处理模式；需要存储的数据大量增加；数据作为取得竞争优势的战略性资产其重要性在增加。

NAS（Network Attached Storage–网络附加存储）即将存储设备通过标准的网络拓扑结构(例如以太网)，连接到一群计算机上。NAS 是部件级的存储方法，它的重点在于帮助工作组和部门级机构解决迅速增加存储容量的需求。需要共享大型 CAD 文档的工程小组就是典型的例子。

NAS 产品包括存储器件（例如硬盘驱动器阵列、CD 或 DVD 驱动器、磁带驱动器或可移动的存储介质）和集成在一起的简易服务器，可用于实现涉及文件存取及管理的所有功能。简易服务器经优化设计，可以完成一系列简化的功能，例如文档存储及服务、电子邮件、互联网缓存等等。集成在 NAS 设备中的简易服务器可以将有关存储的功能与应用服务器执行的其他功能分隔开。

这种方法从两方面改善了数据的可用性。第一，即使相应的应用服务器不再工作了，仍然可以读出数据。第二，简易服务器本身不会崩溃，因为它避免了引起服务器崩溃的首要原因，即应用软件引起的问题。

NAS 产品具有几个引人注意的优点。首先，NAS 产品是真正即插即用的产品。NAS 设备一般支持多计算机平台，用户通过网络支持协议可进入相同的文档，因而 NAS 设备无需改造即可用于混合 Unix/Windows NT 局域网内。其次，NAS 设备的物理位置同样是灵活的。它们可放置在工作组内，靠近数据中心的应用服务器，或者也可放在其他地点，通过物理链路与网络连接起来。无需应用服务器的干预，NAS 设备允许用户在网络上存取数据，这样既可减小 CPU 的开销，也能显著改善网络的性能。

NAS 没有解决与文件服务器相关的一个关键性问题，即备份过程中的带宽消耗。与将备份数据流从 LAN 中转移出去的存储区域网（SAN）不同，NAS 仍使用网络进行备份和恢复。NAS 的一个缺点是它将存储事务由并行

SCSI 连接转移到了网络上。这就是说 LAN 除了必须处理正常的最终用户传输流外，还必须处理包括备份操作的存储磁盘请求。

技术，我们先了解一下几个相关技术的发展过程首先是并行 SCSI 的发展过程。

首先是并行 SCSI 的发展过程。

名称	标准规范	出现年份	理论带宽	关键特性
SASI		1979		SCSI 雏形
SCSI-1	SCSI-1	1986	~2MB/s	异步；8 位
SCSI-2	SCSI-2	1989	10MB/s	同步；16 位
名称	**标准规范**	**出现年份**	**理论带宽**	**关键特性**
SCSI-3	分离指令集，传输协议与物理接口脱离，成为独立标准			
Fast-Wide	SPI/SIP	1992	20MB/s	
Ultra	Fast-20 附加标准	1995	40MB/s	
Ultra2	SPI-2	1997	80MB/s	低压差分机制
Ultra3	SPI-3	1999	160MB/s	附加校验
Ultra320	SPI-4	2001	320MB/s	分包机制；QAS

1.13.3SAN-存储区域网络

SAN(存储区域网络)通过光纤通道连接到一群计算机上。在该网络中提供了多主机连接，但并非通过标准的网络拓扑。

关于 SAN，在 1.7 章节部分有详细说明，这里不作详细说明。

1.13.4SAS-串行 SCSI 技术

SAS（Serial Attached SCSI）即串行 SCSI 技术，SAS 磁盘已经成为主流磁盘类型。

什么是 SAS?简单的说，SAS 是一种磁盘连接技术。它综合了现有并行 SCSI 和串行连接技术（光纤通道、SSA、IEEE1394 及 InfiniBand 等）的优势，以串行通讯为协议基础架构，采用 SCSI-3 扩展指令集并兼容 SATA 设备，是多层次的存储设备连接协议栈。为了更好的了解 SAS

正是 SCSI-3 指令集的出现，使得 SCSI 通讯出现了分层结构，并使 SCSI 指令通过其他物理媒介传输成为可能。事实上，SCSI-3 指令自诞生之日就被一批新技术相中，此后出现的光纤通道技术、SSA 技术、IEEE1394 火线技术等，均受益于这一进步。

这些串行技术虽然从名字上看与 SCSI 毫不相干，但其实它们都支持 SCSI-3 作为应用层逻辑指令。下面是这些串行技术的简要回顾。

名称	标准规范	出现年份	理论带宽	关键特性
光纤通道	FCP	1995	100MB/s	支持光介质
SSA	SSA-S2P/TL1/PL1	1996	20MB/s	IBM 独有技术
SSA	SSA-S3P/TL2/PL2	1997	40MB/s	IBM 独有技术
火线（IEEE1394）	SBP-2	1998	50MB/s	
光纤通道	FCP-2	2002	200MB/s	
InfiniBand	SRP	2002	250MB/s	4 倍速和 12 倍速
iSCSI	iSCSI	2003	~100MB/s	

除了并行 SCSI 和几种应用 SCSI-3 指令集的串行技术，
我们再简单回顾一下 ATA 技术的历史。

名称	标准规范	出现年份	理论带宽	关键特性
IDE		1986		未标准化
	ATA	1994		PIO 模式 0/1/2
E-IDE	ATA-2	1996	16MB/s	PIO 模式 3/4，LBA
	ATA-3	1997	16MB/s	S.M.A.R.T.技术
	ATA/ATAPI-4	1998	33MB/s	冗余校验
UltraDMA66	ATA/ATAPI-5	2000	66MB/s	UltraDMA 模式 3/4
UltraDMA100	ATA/ATAPI-6	2002	100MB/s	UltraDMA 模式 5，48 位 LBA
UltraDMA133	ATA/ATAPI-7	2003	133MB/s	UltraDMA 模式 6
SATA	ATA/ATAPI-7	2002	150MB/s	串行 ATA
SATA-II	ATA/ATAPI-8	2004	300MB/s	优化指令队列

以上三个表格，因为 SAS 技术是以串行机制为基础，对
SAS 这种技术的了解，不是很短篇幅就可以介绍清楚的，
这里只是给我们用户一个初步的认识。

2 LanderCluster 概述

2.1 LanderCluster 的版本情况

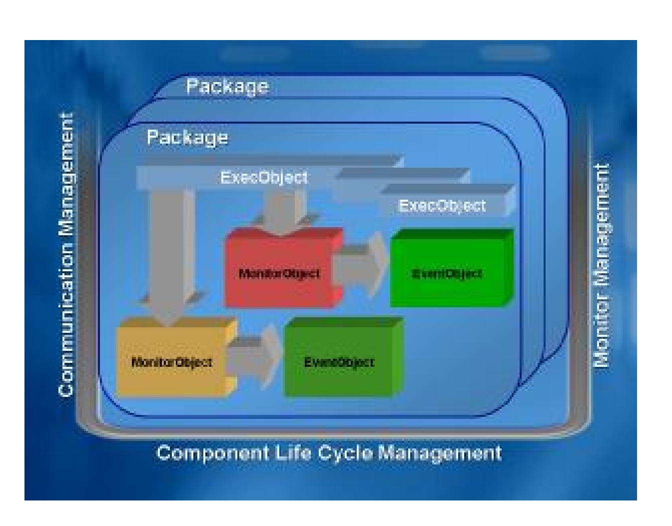
LanderCluster 系列产品是来自中国本土的，拥有自主知识产权的多机高可用产品，支持 Windows、Linux、SCO Unix、Solaris 等主流平台，支持几乎所有的存储环境。集群产品系列分为 LanderCluster-DN 和 LanderCluster-MN。

LanderCluster-DN 双机高可用产品：是联鼎软件产品线中的品牌产品。支持 Windows/Linux/SCO Unix 等操作系统平台，具有稳定可靠、易于管理、开放性强、性价比高的特点，同时支持多语种、支持远程管理等功能，是您构建高可用环境的必要选择。

LanderCluster-MN 多节点集群产品：是联鼎软件产品线中的旗舰产品。支持两个及两个以上节点的集群环境，支持 Windows /Linux/SCO Unix 等操作系统平台，具有稳定可靠、易于管理、备援方式灵活多样、节点和任务伸缩性强、保护用户投资、整合和优化用户系统环境、支持包括 ISCSI 存储环境等特点。该产品最多能够支持 64 节点的集群环境。

2.2 LanderCluster 的体系结构

Cluster 的体系结构，以面向“对象”为核心，对象包括：IPAddress，Alias，Volume，Process，Service，CPU，Memory，Network 等，每个对象都有自己的属性、方法、事件。集群容器（Container）是一个大的接口池，他负责管理这些对象的生命周期，为每一个对象提供接口服务。



Cluster Architecture

2.2.1 执行树模型（Execute Tree Model）

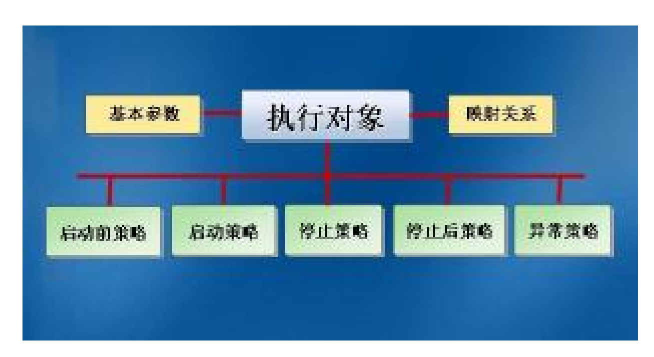
执行树包括系统对象树（Sys Tree）和应用对象树（App Tree），对于集群资源包（Package）而言，系统对象树是固定的，不存在启动和停止接口，而对于应用对象树，情况要复杂许多。关键业务系统，对于资源的启动次序，是有逻辑依赖关系的，因此，资源启动的步骤，必须严格按照执行树定义的 Step 来执行。



Execute Tree Model

2.2.2执行对象模型

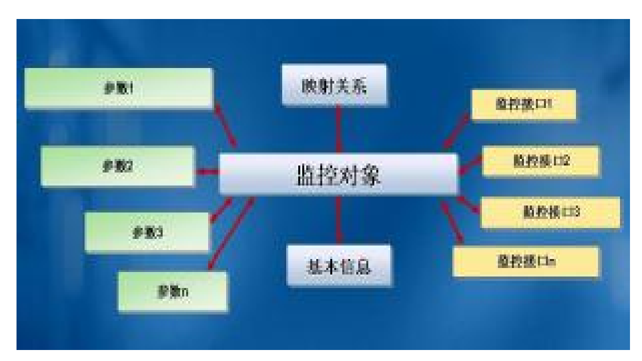
Execute Object Model



Execute Object Model

2.2.3监控对象模型

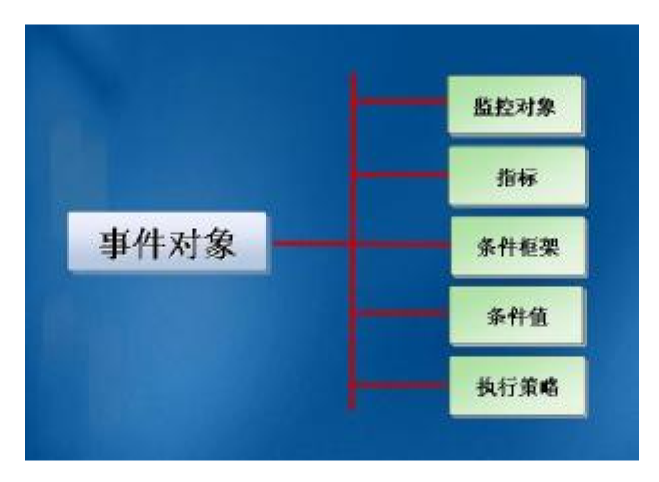
Monitor Object Model



Monitor Object Model

2.2.4事件对象模型

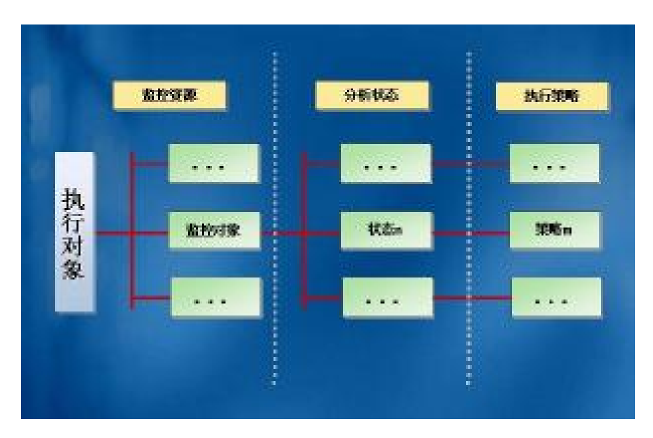
Event Object Model



Event Object Model

2.2.5集群原理模型

Cluster Elements Model



Cluster Elements Model

2.2.6关键技术

A、MLDC 多链路数据交换协议（MLDC Protocol）

MLDC 协议是多机集群环境中节点间的数据交换协议。支持的链路包括心跳网卡、工作网卡和共享存储交换区。集群在工作时，参照链路优先级和节点的实时状态，选择可用链路与其他节点间交换信息，保障集群管理节点能够准确获取各种资源信息。该协议传输可靠，效率高，在多节点环境下性能平稳。

B、存储控制引擎（Storage Agent）

存储控制引擎是集群在实现非共享集群技术的核心组件，在确保数据安全的情况下，能够快速在节点间交换控制权，解决数据的读写一致性问题。该引擎支持主流的存储厂商硬件平台，支持 NAS、SAN 等网络存储环境。

C、应用保护模块（Application Plugin）

集群是从系统层面来保护计算机资源的高可靠，保护的资源包括：CPU、内存、网络流量、IP 地址、虚拟主机名、共享存储、系统服务、进程等。通常，如果这些资源是可靠的，那么关键业务系统就是可靠的，LanderCluster 的专业应用插件，能够从应用层实现对可用性的辅助侦测，能够模拟业务终端对服务器的访问，能够通过应用接口采集应用自身的工作性能，集群通过各种应插件，就能够准确的感知系统的潜在风险，使集群对业务系统的可用性判断更加精准，同时具有预见性。

2.3 LanderCluster 集群系统工作原理

LanderCluster 集群系统软件是由三部分组成，这三部分协同工作，共同完成主机系统的备援工作。

LanderCluster 软件在启动时，首先读取集群系统的配置文件，在该文件中描述集群系统中各节点的网络信息，硬件描述以及任务的定义等参数。

集群核心程序根据集群的配置信息，进行集群系统的状态重组。根据当前的网络状态和集群参数，对节点中的服务器进行调整，建立集群的初始状态。

在节点初始状态建立起来后，根据规则网络管理模块向管理模块提交各节点的网络状态，管理模块根据各节点的网络状态和集群中对资源的定义，对集群中的各节点进行网络资源分配，使集群中的某个节点获得对外提供网络服务的资源。

同时启动节点监控功能，对集群中的节点进行网络状态监控，保持网络状态的健康。

当集群管理模块对网络资源进行分配后，通过任务管理模块对集群中的任务进行分配，根据集群网络资源的分配情况，将与该网络资源相依存的任务分配给已获得对外提供网络服务的节点。

集群任务启动后，该模块启动任务监控功能，对所启动任务的关键进程进行监控。保障对外提供服务的资源健康。

当以上资源建立起来后，集群系统进入正常运行状态。

LanderCluster 高可用集群系统，在进入正常运行状态后，通过专用的通讯链路和集群中的其它节点进行通讯，传输各节点的状态信息，使各节点的核心管理模块获得整个集群节点的实时状态。

当系统中有节点故障时，集群管理模块根据集群当前的状态和该故障节点在集群中的角色做出集群系统是否重组。当该节点为生产机时，集群系统会自动将属于该节点的资源和任务移交到下一个备用节点。保证该业务正常运行。

如果该节点为备份服务器，则需要通知整个集群对备援状态进行调整，将该故障节点从备援设备表中删除。使备援记录中不再有该故障节点的记录。直到该节点修复后重新在线，集群软件自动进入集群中作为备援节点角色。

2.4 LanderCluster 集群系统监控原理

当集群系统正常运行后，LanderCluster 高可用集群系统进入系统监控状态。在监控状态下具有网络状态检测、应用程序检测、集群软件自身状态检测和存储子系统检测。

2.4.1网络状态监控

在集群运行中，LanderCluster 集群软件的网络管理模块对整个网络中的网络资源进行实时监控，获取整个网络的运行状态。

如果监控到集群中有节点失效时，将该节点的网络状态通知到 LanderCluster 的管理模块，管理模块根据当前的网络状态和该节点在整个集群中的角色（生产机或备援机），通知整个集群进行状态重组。

如果该节点为生产机，则管理模块通知集群中的下一个备援服务器进行任务接管。从集群中剔出该故障服务器。对整个集群重新分配规则。

如果为备援机，则管理模块通知整个集群节点进行规则调整，将该故障节点从备援节点表中剔出，保持集群系统中节点的有效性。

2.4.2应用监控

对外提供服务的应用程序一般为数据库或中间业务系统，如果应用程序出现故障，则集群中的该节点也无法正常提供对外的服务。

为提高集群的可用性，LanderCluster 可以在集群资源中灵活定义对进程进行监控的方式。对进程名监控还是对进程个数进行监控。

当关键进程丢失或进程个数达不到一个固定的阀值时，集群会将资源转移到下一个节点运行，保障应用系统的正常运行，保持整个集群的健康状态。

2.4.3集群软件运行状态监控

集群软件在运行的过程中，因其它因素的影响，会造成自身的进程丢失。如果自身进程丢失，会影响到整个集群的运行状态。

LanderCluster 集群软件，实现对自身进程的监控，当人为或意外操作将某个 LanderCluster 的服务进程退出运行时，LanderCluster 会自动将该丢失的进程重新运行。保障 LanderCluster 系统的自身运行安全。

2.4.4存储子系统监控

在集群系统运行过程中，所有的数据均存放在共享的磁盘阵列子系统中，当磁盘阵列子系统因连接线或 SCSI 卡出现故障导致无法对主机提供服务时，LanderCluster 高可用系统根据配置的集群资源，并确认当前的主机是工作机（生产机）时，会自动将该主机的任务移交到备用节点，使系统可以继续服务。同时在日志中报警，提醒用户对该故障进行处理、维护。

LanderCluster 高可用集群软件，通过集群节点间的心跳信号，和其它节点进行通讯，获得其它节点的运行状态，根据整个集群中各节点的状态，更新本节点自身的状态表。同时根据集群管理层的命令，调整自身节点的状态和集群资源。

LanderCluster 集群系统软件通过实时对集群系统资源的监控，及时发现集群中节点的故障，及时通过备用节点代替故障节点的工作，使集群状态处于一个完整的健康状态。

2.5 LanderCluster 集群系统的特性

2.5.1系统健康与可用性评价体系

一个核心业务系统的可用与否的关键因素是整个系统的健康程度，传统高可用系统仅仅简单的认为系统只有“可用”与“不可用”两种状态，这样的判断虽然简单但却仅仅考虑了两种极端的状态，显然无法对系统进行全面保护。LanderCluster 6.0 在业界首次提出了创造性的“系统健康评价体系”并加以应用，我们认为对系统的可用性判断必须是持续的，而非极端的，大量的情况证明系统从“可用”转向“不可用”并非是瞬间发生，而存在一个过程，在这个过程中，系统的某些核心指标将会提前显示出异常，虽然此时系统对外表现出的仍然为“可用”，但整个系统实际已经处于“亚健康”，对客户端的请求反应逐渐变慢，系统出现不稳定的迹象，系统整体可用性逐渐降低，如果不进行任何干预，系统在未来某个时点将有极大可能转化为真正“不可用”，从而导致灾难性的后果。通过 LanderCluster 6.0 的“系统健康评价体系”全新的系统核心指标持续检测功能，用户将及时发现系统的异常状态，有效判断系统目前真正的“健康”程度，并且经过对系统核心指标的综合分析，将能对系统未来可能发生的状况进行“预知”，直击造成系统转向“不可用”的原因，提前发现，提前预警，提前解决，令用户从“被动”的解决转向“主动”的发现与处理，让系统的可用性判断从“不可知”转化为“可预知”。

2.5.2前瞻预警体系

当系统完全瘫痪时再进行拯救，犹如为心脏停止跳动的病人进行复苏，难道不觉得太晚了吗，如果我们能够预知系统将逐渐变得不稳定，而提前做出应对，防止系统崩溃，或者将突发性宕机转化为计划性维护，将对您产生更多益处。LanderCluster 全新的系统智能预警体系，持续监控维持核心系统稳定运作的重要指标变化，包括处理器、内存、LAN 介质、存储设备、网卡、进程、应用程序实时状态，任意指标出现异常状况，即可快速做出响应，防患于未然。

2.5.3故障分级处理

传统集群软件仅将系统宕机定义为故障，然而“故障”就仅仅是“宕机”，无法访问吗？真实的“故障”应当以系统健康状况及用户的承受能力作为衡量标准，不同的用户对系统故障的定义是不同的，LanderCluster 独特的故障分级处理系统能够满足用户自定义故障阀值，建立不同的故障评价标准，并对每一类故障进行自定义操作，提供最大的灵活性。同时系统提供丰富插件，为用户提供精准的故障分析。

2.5.4深度应用侦测代理（User Application Agent）

集群保护下的核心业务，是通过代理（Agent）实时采集应用的运行态数据，结合“评价体系”来诊断系统可用性的。可用性指标分为两类：一类是结果类，即模拟客户端访问是否成功，是否获得期望的响应；另一类是风险类，体现的是系统当前运行态的风险指数，如应用的连接数、数据库的存储空间使用率、Web 的访问迟延、网络的流量、CPU 的负载、系统内存的余量等等，这些因素都是系统能否正常工作的潜在风险，是进行故障预警的重要预测依据。

LanderCluster 提供常用软件的侦测代理，如 Oracle、MS－Sql 以及 Web 等，这些监控对象的接口及方法，被灵活保存在 XML 配置文件中。集群提供开放的应用代理接口，用户可根据开发模板，自定义监控对象的指标采集方法，就可以让集群系统实时监控这些指标，触发相关的事件。

2.5.5支持虚拟化环境

虚拟化大潮翻涌，软硬件虚拟化技术不断扩展，用户未来的核心应用极有可能运行于虚拟化的环境中，在虚拟化技术重新整合并分配用户资源的同时，对系统整体可用性的要求变得更为苛刻，一台运行了多个虚拟环境的服务器一旦出现故障将直接影响其上的所有虚拟系统，其损失远比单一系统环境严重数倍。单个虚拟系统的故障同样需要进行故障转移，LanderCluster 超前支持虚拟化存储环境，支持虚拟化操作系统环境 VMware，支持单一虚拟化系统之间自由切换。

2.5.6采用任务提交、确认机制

在集群系统中，节点之间通过消息确认方式进行任务的移交。主服务器在对任务进行移交时，对任务进行关闭后，通知备份服务器进行任务接管工作。当备份服务器没有接到确认消息时，始终处于等待状态，直到接到确认消息。

当备份服务器在长时间没有接到确认消息时，会通过侦测对方的任务状态来判断，主机的任务是否安全关闭。如任务已关闭，则通知主服务器要接管任务，并开始执行任务接管。如果任务没有关闭，则主服务器处于僵死状态（操作系统故障）时，对该任务进行强制接管。并通知集群系统该主机不可用。

LanderCluster 高可用集群软件，对集群节点中的关键操作均采用确认方式，确保任务安全移交，杜绝双主机、多主机状态和双任务状态。

2.5.7集群配置安装维护简单

图形化的配置管理界面，对集群文件系统配置、网络配置以及任务的配置方式均通过选择方式进行，操作简单易用。

LanderCluster 高可用集群软件采用简洁的菜单选择方式，对集群中的资源进行配置，不采用编写脚本的方式进行配置，而是在每项菜单中对集群的资源配置以表格的方式进行填写。使软件具有很好的可用性。

同时通过菜单和表单方式进行组合，使管理员对集群的配置维护都具有很高的直观性。使软件便于设置和维护。

2.5.8管理员密码验证

对于一个集群来讲，该集群的资源配置参数最为重要，任意修改该资源配置参数，则会导致整个集群的运行。

LanderCluster 高可用集群软件为保护集群配置的安全，在对集群资源配置时，增加了用户口令验证，只有持有该口令的管理人员才能对集群的资源进行修改、配置。通过口令验证方式，对集群的配置安全做进一步的保护。

2.5.9集群软件自身监控功能

集群的安全的另外一个重要的因素是自身的安全，当程序因意外故障导致集群服务主程序退出时，需要能够对退出的运行程序进行处理。

LanderCluster 高可用集群系统采用自身监控的功能，当某个程序退出运行时，集群自身能够对该退出的程序进行重新启动，保护集群软件健康运行。

2.5.10对应用程序的灵活监控功能

LanderCluster 高可用集群软件在对应用程序监控时，采取非常灵活的方式。可以对关键进程进行监控，也可以对进程的个数进行监控。当定义对进程个数进行监控时，

只需要对监控的进程个数设置一个阀值，当进程个数低于该阀值时，系统会自动发送通知到管理核心模块，对该任务进行任务移交。

2.5.11支持多台服务器集群方式

LanderCluster 支持从两个节点的简单集群系统，平滑过渡到以后的多节点集群系统，对业务系统整合以及优化业务系统有很大的优势。

2.5.12支持远程管理模式

LanderCluster 采用流行的 C/S 方式对集群进行管理、维护及其监控等操作，均可以通过客户端方式进行操作，不需要在服务器上进行操作。客户端通过直观的图形方式对集群的整个状态进行实时监控。当集群有故障时，客户端通过声音、邮件方式进行报警处理。并在集群的监控窗口显示故障点的位置。

2.5.13集中管理

LanderClsuter 内置远程集中管理工具可以使让您方便的对系统中的多个集群子系统，包括 Linux、Windows、AIX、HP-UX、Solaris 等所有支持的操作系统 LanderCluster 集群环境进行统一、集中的管理。大大降低管理复杂度，减少管理成本。

2.5.14支持更多存储环境 包括 ISCSI

集群软件采用系统级的硬件处理，与硬件无关性，只要操作系统支持的硬件、LanderCluster 集群软件均可以支持，支持流行的 SAN 架构的光纤磁盘阵列子系统、SCSI 结构的磁盘阵列子系统、以及 ISCSI 存储环境。

2.5.15支持多种应用系统

支持目前流行的数据库系统，如 Oracle、Sybase、MySql、MS SQL Server、DB2 等几乎所有数据库环境。

支持应用系统：Microsoft IIS、Apache、WebLogic 等应用服务器环境。

支持群件系统：IBM Notes 等。

支持各种第三方应用系统，如用友财务软件、SAP 等。

2.5.16LanderCluster 灵活性

随着用户机构不断扩容，核心业务不断增多，用户将对整体系统抵御灾难的能力提出更高的要求，单纯的局域网环境内高可用系统将扩展为远距离的广域网环境高可用系统，成为容灾体系的重要环节。LanderCluster 完全支持广域网环境，并增加了 CheckPoint，能够搭建局域网与广域网共存的复杂环境，为用户提供更可靠的保障。

2.5.17中英文管理界面 可根据需要选择

2.5.18创新支持跨平台系统集群

LanderCluster 独特创新的 MLDC 控制协议，使得我们可以在异构操作系统环境下进行统一管理和故障切换。结合数据库、应用层面的复制，可以实现多种低成本、高效的系统业务容灾，甚至无需应用迁移，就可以实现跨平台的集群实现。比如，Windows 环境下 Apache 应用，通过 LanderCluster，可以切换到 Linux 操作系统下的 Apache。

2.6 LanderCluster 版本比较

功能特性	LanderCluster-DN	LanderCluster-MN
支持节点数	2	64
任务数	2	64
支持插件数	2	4
存储控制	SAN(FC、ISCSI)/DAS/NAS等	SAN(FC、ISCSI)/DAS/NAS等
扩展模块支持	Replicator	Replicator
支持Replicator节点数	2	16
支持Replicator方向	双向	多向
扩展性	好(强)	好(强)
可靠性	高	非常高
通信链路	3	4
备援方式	Active/Standby：主备 Active/Active：主主 单机高可用	Active/Standby：主备 Active/Active：主主 N à 1：多备1　1 à N：1备多 N à M：多备多　N à N：多机互备 单机高可用
集群自监控	支持	支持
参考点监控	不支持	不支持
前瞻预警	主机CPU监控，内存监控 网络负载监控，进程占用CPU监控 进程占用内存监控	主机CPU监控，内存监控 网络负载监控，进程占用CPU监控 进程占用内存监控
故障分级	自定义集群事件，声音/邮件	自定义集群事件，声音/邮件
远程监控	支持	支持
集中平台管理	支持	支持
中文/英文界面	支持	支持
专业应用保护模块	可选	可选
任务负载均衡	支持	支持（可设置）
安全认证	高	极高（RSA1024位加密）
事件日志	详细	详细（支持调试模式）
协议	MLDC	MLDC

3 LanderCluster 规划技术要点

3.1 LanderCluster 硬件配置概述

手册的第一部分介绍了有关集群概念及如何避免单点故障的手段。配置高可用系统的目的是保证系统可以不间断的提供服务，因此硬件配置的一个关键是尽可能的减少单点故障，而手段主要有两种：使用容错服务器和配置集群环境。我们这里仅讨论通过 LanderCluster 实现的集群高可用环境，因为容错机的普遍使用还不现实，价格昂贵加上维护困难。目前国内外用户普遍采用的是集群环境，占大多数的是双机集群，国内通常称为双机容错。

联鼎拥有从高端小型机到 Intel 构架服务器环境下的 Windows Server、SCO OpenServer/UnixWare、Linux 等的集群及双机容错解决方案。

高可用集群环境下的双机或多节点高可用并非完全没有单点故障，就像前面介绍的，完全没有单点故障的环境是没有的。在我们的 LanderCluster 集群环境下，通过合理配置硬件设备，可以尽量减少单点故障点。下面我们讨论几种建议的配置环境和设备选型的原则。

为提供高级别的可用性，典型群集软件使用冗余系统组件，如采用两个独立的磁盘等方式提高系统的可用性。这种必要的硬件冗余结构主要是消除整个系统的单点故障。

一般来讲，冗余程度越大，出现故障时访问应用程序、数据和支持性服务的可靠性就越大。除硬件冗余外，系统还必须具有软件支持，因为软件支持在出现故障后启动和控制应用程序向另外一个网络或节点进行转移。LanderCluster 就是基于这样的需求由联鼎软件自主研发的集群软件包，可以提供以下支持：

A　在网络出现故障的情况下，LanderCluster 自动将受到影响的任务转移到备用节点。

B　在其它受集群系统管理的资源出现故障的情况下，LanderCluster 自动将程序转移到备用节点上。

C　在软件出现故障的情况下，应用程序可以在另外一个节点上重新启动，针对整个系统来讲，同时中断的时间最短。

通过 landerCluster 构造的高可用系统，使您具有对硬件系统进行

在线升级的功能，通过 LanderCluster 可以轻松的将系统转移到另外一个节点上，以便对当前的系统进行维护和升级等操作，当系统升级结束后，再将任务移交至本机，再对另外一个节点进行维护和升级。

3.2 集群设备选型要点

我们前面有很多关于单点故障的描述，那么在配置集群环境中我们如何选择自己的硬件环境呢？

在具体表述之前，我们先回顾一下一个高可用集群系统的可用性是如何得到的。一个双机环境通常由两台服务器和一个磁盘阵列，通过一个 SCSI/FC 链路串接在一个“总线’上，那么其整体的可用性等于 ：服务器 1 的可用性 X 服务器也就是说如果服务器 1 是 99.99，服务器 2 是 99.999，磁盘阵列是 99.99，则双机（对等工作方式：Active/Active）环境下的整体可用性是：

0.9999X0.99999X0.9999=0.99

而对于一个磁盘阵列子系统来分析其可用性，则是由组成阵列的各部件的可用性相乘得到。磁盘阵列系统由电源、背板、控制器、多个硬盘组成，那么磁盘阵列整体的可用性通常达到五个九（99.999）已经很难了，因为很多磁盘阵列都是盘和柜单配的，而市场上的硬盘通常可靠性不是很高的。导致整个磁盘阵列可靠性不高，从而最终导致整个高可用环境的可靠性低。

因此我们得到的结论是高可用环境下的硬件设备选择是系统的可用性的基石，而一个好的高可用软件使集群成为现实。

选择硬件的原则可以归纳为：

A　性价比是选型的要点；

B　磁盘阵列是集群系统的核心，它的可靠性是关键，

性能次之；

磁盘阵列有可能是单点故障点，它必须在环境中具有最

高的可用性和可靠性；

C　集群中的服务器可以选择不同配置，但尽量相同品牌；

D　尽可能少的留有单点故障点；

E　选择开放性好的服务器，可靠性尽量的高；

F　尽量采用设备独立的存储子系统，既尽量采用带有独立 RAID 控制器的存储设备；

G　选择磁盘阵列尽量考虑其硬盘的可靠性，最好和阵列柜统一考虑；

H　磁盘阵列一定要求双路电源、散热性、抗震性、抗干扰能力等都是很重要的；

I　正确认识磁盘阵列控制器的有关参数：CPU、Cache、通道等,这些数值不能代表磁盘阵列的可靠性和性能的高低。

分析：

单独就双机环境下的集群来分析，设置两台服务器的目的是使系统处理单元（SPU）达到冗余，而共享存储冗余的代价太大，一般选配一个单柜来实现，而 LanderCluster 软件包负责监控系统，并在系统故障时报警并做出相应的切换操作，保证服务不丢失。但服务不丢失的前提是磁盘阵列部分不出故障，一旦磁盘阵列的控制器故障导致设备无法访问，则无论服务器主机有多好的性能和可靠性，都无法阻止系统停止服务。选择可靠磁盘阵列是关键。

如果真的要做到没有单点故障，则可以配置双控制器的磁盘阵列来避免控制器损坏导致的宕机，也可以配置双柜来达到磁盘阵列的冗余。两种方式各有利弊，双控并非完全避免单点故障，它涉及到控制器热切换，而且两个控制器同时在线，同时损耗，也无法避免由于硬盘损坏导致的停机；双柜方式通常需要特定软件的支持，保证磁盘柜之间数据的同步，同时代价很大，这两种情况要根据实际需求来定义。

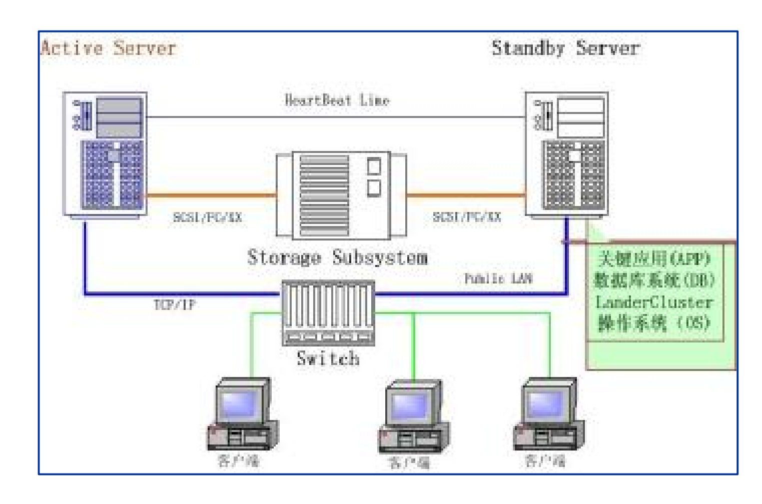
服务器是服务提供的运行部分，它应该有很高的可靠性和开放性，便于维持整个系统的扩展性和开放性。集群环境中的服务器可以配置不一样，可以节省投资，因为 LanderCluster 双机用户中，通常采用的都是主从（Active/Standby）工作方式，备份服务器绝大部分时间是等待状态，可以配置的比主服务器低。

集群环境中的应用数据，要求存放在共享磁盘阵列中，本地硬盘通常安装操作系统、应用软件及 LanderCluster 软件包，为保证主机系统安全性，建议系统盘通过 RAID1 实现镜像，保证一块系统盘的损坏不会导致系统服务切换或终止。

3.3 LanderCluster 简单双机集群环境

简单双机集群是目前大多用户采用的高可用环境，简单的说就是两台服务器加一台磁盘阵列，通过 LanderCluster 软件实现主从工作方式的双机环境。这样的环境不一定是十分严格的集群，因为按照前面描述的有关内容，需要考虑的问题太多，包括硬件配置、单点故障等。

主从就是热备工作方式，容错软件作为不可缺少部分起到监控系统状态并在系统故障时，自动做出相应的反应，保证整个系统提供服务的不间断。对于这样的环境这里不进行过多的描述，理解下面的示意图就可以了。



3.4 LanderCluster 复杂双机集群环境

复杂双机集群环境通常指针对特殊用户需求而实现的较简单双机复杂得多的双机应用环境。这样的环境包括对

等双机、双机双柜方案、异地双机（带容灾功能）等，这些环境的实现是有一定技术难度的。下面我们分别讨论几种复杂双机的定义。

3.4.1对等双机（Active/Active）

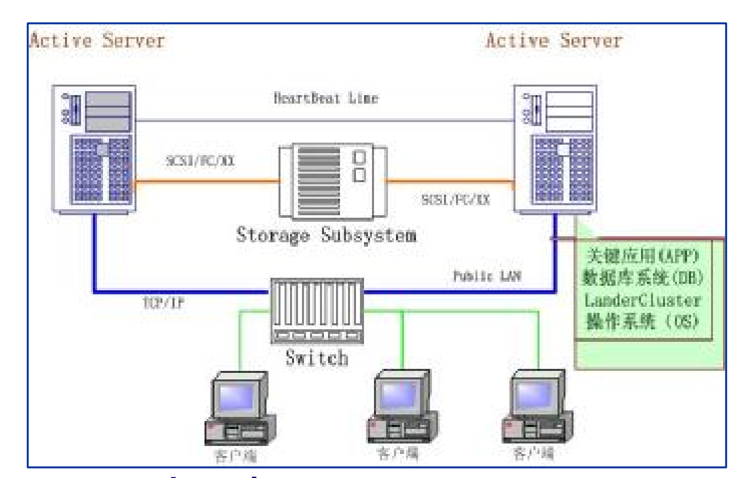
对等双机就是我们通常说的双主机工作方式，这个环境下，有两套不同的应用运行在集群环境中，每台服务器运行各自的应用，在其中一台出现故障时，另外服务器将接管其服务。这种配置可以大大提高设备的利用率，缺点是增加了系统的复杂度，而且对于某些特殊应用环境可能无法实施。

对等双机是真正的双机互备，要求服务器具有较强的处理能力，来满足两个应用的需求。在配置对等双机时，硬件的配置与主从双机略有不同，主要在网卡上。对等需要至少两片网卡，每个网卡对应一个应用，可以是相同或不同网段的网络地址。

对等双机通常要求两个应用的共享存储部分完全独立，它不同于通常的并行服务器，也不具有负载均衡功能。并行服务器典型的是 ORACLE 的 OPS（Oracle Parallel Server）,它是多台服务器运行一个 ORACLE 数据库，通常需要特殊的底层软件包来支持，因为并行服务的关键是硬盘访问的一致性控制，ORACLE 上称之为 DML（分布锁管理），来控制数据的访问。目前，这样的环境只能在某些特殊的高端环境下运行。LanderCluster 还做不到并行服务器的控制功能。

而负载均衡的概念是对访问或处理的动态资源分配，保证最大限度的使用硬件资源。也有这样的原因使用负载均衡：访问或处理集中在一个数据资源上，而运行该数据资源的服务器根本无法满足访问的需求,这时需要多个服务器来接受访问，那么动态的将访问需求分配到不同的服务器上，来满足需求。

总之，LanderCluster 在对等方式下，满足的是对两个独立的应用实现高可用的需求。理论上我们可以将多个不同应用分布在两台服务器上，使得多个应用可以在高可用环境下运行，这样可以达到多应用互相备援的目的。因为 LanderCluster 可以对进程监控、对进程数量监控，对 LanderCluster 来讲，本身不区分进程的类型，仅仅把进程作为监控的对象而已。下面是 LanderCluster 对等方式的图解，请仔细理解。



3.4.2双机双柜

双机容错系统的最高境界是完全避免单点故障，而没有单点故障的系统是几乎不存在的。但是，双机环境中的存储部分的重要程度我们前面有大量描述，因此，双机双柜作为一种可选方案在某些环境下是可行的，而且具有很多的优点。

下面针对 LanderCluster 环境下，一个 SCO UNIX 双机双柜系统，系统没有物理单点故障（存储部分）。LanderCluster 环境经过测试，在任何一种情况下都能保证系统不间断运行。

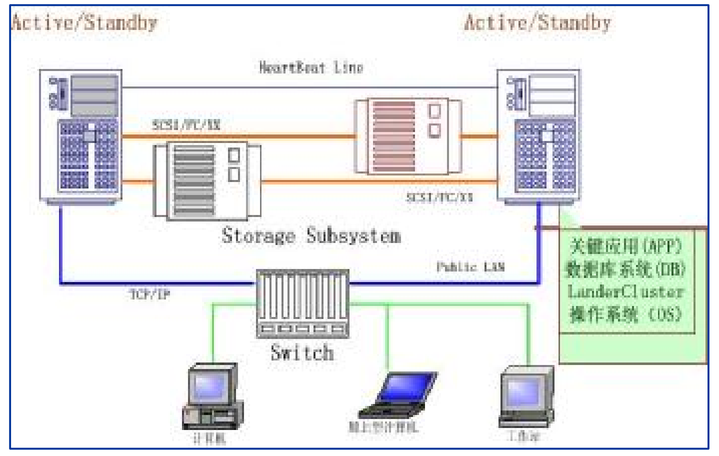
运行主机不宕机，有一个磁盘失效。此时运行主机上数据库能正常运行，运行主机也能正常运行，而备机也不受干扰。此时，整个系统环境能正常运行。

当运行主机宕机，一个磁盘失效时。备机能顺利接管数据库的应用。接管后，数据库能正常运行。在此情况下，整个系统环境也能正常运行。

所以，在采用双机双柜用数据库作镜像应用情况下，基本达到了能够保证数据库应用不会由于单点故障（单个磁盘柜或单个主机失效）而造成整个系统环境的失效。

双机双柜的实现是有很多的前提的，其中最关键的是磁盘柜数据的同步问题。由于双柜环境实际是磁盘阵列的冗余，而冗余的盘阵必需具有同样的数据内容，就是说要求数据在写入磁盘阵列时，必须同时写两个盘阵，否则该设备冗余没有意义了。在很多环境下是不能运行双柜环境的，但在 Informix 环境下，通过 Mirror Chunk 可以通过数据库本身达到 镜像功能。而 Oracle 等一些数据库就无法实现了。

因此，针对本环境需要具体问题具体分析，可将应用需求通过邮件或电话提交给联鼎软件集群支持中心确认方案的可行性。



3.4.3异地双机（容灾）

是利用光纤存储技术特点实现的一种双机环境。这种方式简单的说是将集群的两台服务器放置在距离较远的地方，使之具有一定的容灾功能。这种情况不是任何应用环境都可以实施的。

硬件上首先需要的是 SAN 结构的存储环境，因为 SAN 存储结构有很好的扩展性、灵活性，同时一般采用光纤作为传输介质，光纤可以在很长的距离内传输数据，使得服务器、存储可以分别放置在距离很远的地点。

本手册第一部分针对一些概念有过描述，容灾和高可用是不同的需求和概念，同时是一个安全的系统应该具备的特点。在能够达到高可用的情况下，如果能在不增加投资的情况下，解决容灾问题，将一举两得。

真正做到容灾必须使用双机双柜环境，而双机单柜也可以成为部分容灾。双机双柜需要特定软件支持，或数据库程序、应用本身支持，而对于没有共享数据的应用系统，这种方式将十分合适。具体环境描述及实施细节，针对不同环境和需求需要进行具体的分析，这里仅介绍存在这样的可能，使您的高可用环境可以更加灵活，解决更多的问题。

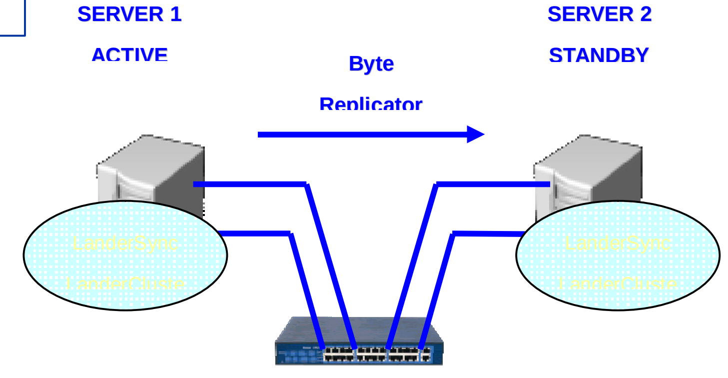
因此，LanderCluster 高可用环境实际上有很多配置方式，看大家的思路有多开阔了。

3.4.4纯软件双机

纯软双机，是指在没有共享存储阵列的情况下，要将 Active 主机上的数据，实时地复制到 Standby 主机，以确保集群切换以后，主备机之间的数据依然是一致的。当服务器对数据的修改频繁发生时，基于文件级的复制就不能满足要求。LanderCluster 配合 Landersync 就能实现高性能的纯软双机。

LanderSync 的复制是字节级的，也就是说，源主机中的某个文件的某几个字节发生了变化，LanderSync 就将这几个字节复制到目标机器中，显然这样的复制方式，要比文件复制效率高得多。

有关 LanderSync 的详细说明，请参考 LanderSync 的相关产品资料。



纯软双机应用

3.5 LanderCluster 多节点集群环境

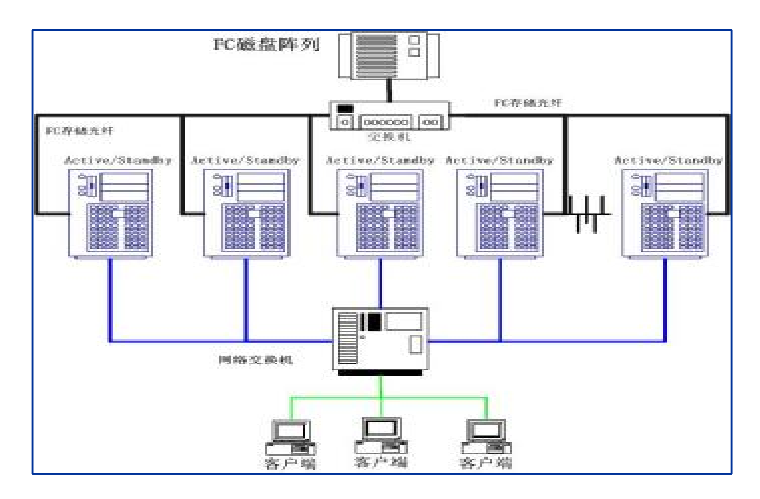
多节点集群在未来会有大量需求，大家对集群的认识目前还在双机环境下，而实际上，在有多个关键应用都具有高可用需求，在一个机房内建立多个双机系统显然是很浪费的，管理的复杂度又高，多节点集群可以有效的解决这个问题。

多节点集群是在存储技术不断发展、用户应用环境日趋复杂的情况下，被提上日程上来。在高端，多节点集群在多年前已经是相当成熟了，而在此环境下，性价比很好的产品还很少。

LanderCluster 就是这样一个非常好的解决方案。这里的多节点集群简单的讲是多节点高可用，可以理解为多机互备，多个服务器连接在一个共享存储设备上，同时运行多个不同应用，在其中任意服务器出现故障时，其它服务器会根据备援规则接管服务，保证整个集群中的服务都能高可用。

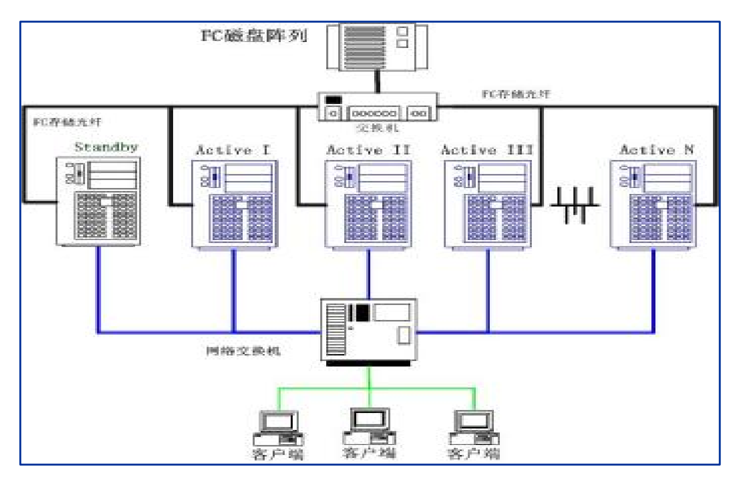
3.5.1多节点 - 多备一

这个方式中多个生产服务器工作在一个集群中，配置一台独立备份服务器，任何一个任务停止工作都将由备份服务器作为第一优先主机接管任务。这种科学的部署方式有效的优化了系统结构，同时使整个集群中应用环境达到高可用。



3.5.2多节点-多机互备

这个方式中多个生产服务器工作在一个集群中，通过合理定义备援规则，使环境中服务器相互备援，任何一个任务停止工作都将由其它服务器接管任务，是一种设备利用率最佳的部署方式。该部署方式同样有效的优化了系统结构，需要注意的是在定义备援规则时，仔细分析各服务器处理能力、任务的兼容性问题。



4 附录

4.1 运行 LanderCluster 的系统需求

硬件要求：

Intel X86（32/64 位）构架服务器产品

外部存储设备（SCSI/FC/ISCSI 子系统，可选项）

服务器配置两个或以上全双工网卡

256M 以上内存

至少一个 Hub/Switch

数据库支持：

SQL Server

DB2

Oracle

Sybase

Informix 等

技术服务：

软件测试中心

800 服务热线

终身免费电话支持

免费远程支持

购买之日起一年免费软件升级

用户产品现场培训、环境优化（限购买安装服务用户）

服务期内有限的现场服务

其他服务参考《标准支持条款》

其他：

用户定制备援规则

用户定制插件

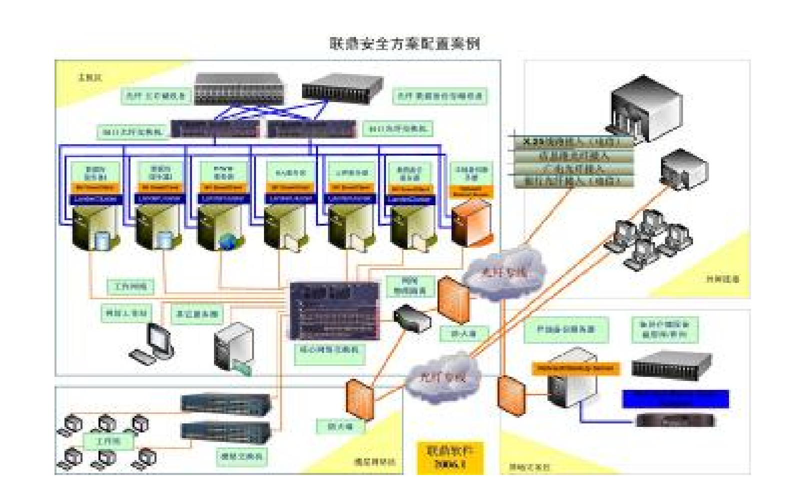
根据用户实际环境定制功能

4.2 典型解决方案

联鼎灾难恢复计划

在金融、电信、政府等行业，系统的高可靠性与高可用性显得尤其重要，核心数据的重要性甚至超过对性能和价格的要求。与此相对应，联鼎软件推出了 LanderVault 将有力保障客户业务系统的持续不间断运转，并避免由于各种不可测因素而导致的核心业务数据的损失。

联鼎软件的灾难恢复技术解决方案囊括了主机、存储、操作系统、中间件、数据库、网络和部分应用系统各个方面。此外，联鼎高素质的支持服务也是必不可少的。从距离和可避免的灾难划分，联鼎拥有本地容灾、园区容灾、城域容灾和跨洋容灾的全面解决方案。为用户的业务持续性提供了有力保障。



灾难隐患与防范

影响业务持续性及造成数据意外丢失的因素有很多，主要包括技术因素、应用因素、人为因素以及环境因素。

技术因素包括：主机硬件故障，os 故障，数据库故障，网络故障和存储系统故障。

应用因素主要是业务应用系统的稳定性和可靠性。

人为因素很复杂，大致可分为内部和外部的，失误的和

故意的。

环境因素包括电源、火灾、建筑物倒塌、盗窃，以及洪水、地震、火山等自然灾害。

以上各种因素引起业务系统停顿及核心业务数据的丢失均可采用联鼎软件提供的专业解决方案进行防范，使损失最小化。

更多解决方案

可以从 **www.landercluster.com** 上获得详细资料

也可以直接致电联鼎软件从销售人员、市场人员处获得。



服务热线：800-820-1776

上海总部	北京分公司	成都分公司	深圳分公司
电话： 021- 64379596	电话：010－51280181	电话：028－85137056	电话：0755-83208930
传真： 021-64671505	传真：010－88554990	传真：028－85150756	传真：0755-83208930
邮编： 200020	邮编：100044	邮编：610041	邮编：518054